## Automatic Image Processing and

## Conversion to Tactile Graphics

A dissertation submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy at Virginia Commonwealth University.

By: Tyler Ferro

B.S. Electrical Engineering Southern Illinois University Carbondale May 2010
B.S. Computer Engineering Southern Illinois University Carbondale

May
2010
M.Sc. Biomedical Engineering

University of Oxford
Sep 2011
Cert. Preparing Future Faculty
Virginia Commonwealth University
May 2018
MBA Information Systems
Virginia Commonwealth University
May 2018
Ph.D. Biomedical Engineering
Virginia Commonwealth University
May 2018

Director: Dianne T.V. Pawluk, Ph.D.
Associate Professor, Department of Biomedical Engineering

Virginia Commonwealth University
Richmond, Virginia
May 2018

# Copyright © 2018 Tyler Ferro 

All rights Reserved

## Academic Use Only

Contact original/referenced author(s) for permission to use material for published/commercial use

## Automatic Image Processing and Conversion to Tactile Graphics

A dissertation submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy at Virginia Commonwealth University.

By: Tyler Ferro<br>B.S. Electrical Engineering Southern Illinois University Carbondale May 2010<br>B.S. Computer Engineering Southern Illinois University Carbondale May 2010 M.Sc. Biomedical Engineering University of Oxford Sep 2011<br>Cert. Preparing Future Faculty Virginia Commonwealth University May 2018 MBA Information Systems Virginia Commonwealth University May 2018<br>Ph.D. Biomedical Engineering Virginia Commonwealth University May 2018

Director: Dianne T.V. Pawluk, Ph.D.,<br>Associate Professor, Department of Biomedical Engineering

Virginia Commonwealth University
Richmond, Virginia
May 2018

## Table of Contents

A Contents ..... A-01
VCU Forms ..... A-01
Copyright ..... A-05
Title Page ..... A-07
Table of Contents ..... A-09
List of Figures ..... A-11
List of Tables ..... A-13
List of Terms ..... A-15
Appendix Contents ..... A-17A-18
B Preface ..... B-01
B. 1 Abstract ..... B-01
B. 2 Project Overview ..... B-02B-04
C Introduction ..... C-01
C. 1 Problem ..... C-01
C. 2 Motivation ..... C-02
C. 3 Core Process ..... C-04
C. 4 Manual Implementation ..... C-05
C.4.1 Modeling ..... C-08
C. 5 Automatic Implementation ..... C-09C-12
D Segmentation ..... D-01
D. 1 Introduction ..... D-01
D. 2 Design Decisions ..... D-01
D.2.1 Features ..... D-02
D.2.2 Classes ..... D-04
D. 3 Design Implementation ..... D-07
D.3.1 K-Means ..... D-08
D.3.2 Level Set ..... D-10
D.3.3 Texture Application ..... D-21
D. 4 Optimization Preparation ..... D-22
D.4.1 Image selection ..... D-22
D.4.2 Professional Diagrams ..... D-22
D.4.3 Novice Diagrams ..... D-26
D.4.4 Professional Rater Reliability ..... D-26
D. 5 Algorithm Optimization ..... D-35
D.5.1 Parameter Tuning ..... D-36
D.5.2 Parameter Selection ..... D-48
D.5.3 Internal Consistency ..... D-50
D.5.4 Algorithm Selection ..... D-51
D. 6 Usability Testing ..... D-54
D.6.1 User Testing ..... D-54D-66
E Simplification ..... E-01
E. 1 Introduction ..... E-01
E. 2 Design Decisions ..... E-02
E.2.1 Modeling ..... E-02
E.2.2 Components ..... E-03
E. 3 Design Implementation ..... E-08
E.3.1 Cue Extraction ..... E-08
E.3.2 Cue/Observation Preparation ..... E-11
E.3.3 Edge Anticipation ..... E-16
E.3.4 Thinning ..... E-19
E.3.5 Region Rejection ..... E-21
E.3.6 Region Cleanup ..... E-23
E.3.7 Accenting ..... E-24
E.3.8 Segment Texture Model ..... E-25
E.3.9 Segment Texture Application ..... E-27
E.3.10 discussion ..... E-28
E. 4 Usability Testing ..... E-29
E.4.1 Image Selection ..... E-30
E.4.2 User Testing 1 ..... E-30
E.4.3 User Testing 2 ..... E-40
E-47
F Postface ..... F-01
F. 1 Conclusions ..... F-01
F.1.1 Segmentation Development ..... F-01
F.1.2 Segmentation User Testing ..... F-02
F.1.3 Simplification Development ..... F-03
F.1.4 Simplification User Testing ..... F-05
F.1.5 Limitations ..... F-06
F.1.6 Population Differences ..... F-07
F. 2 Future work ..... F-07
F-10

## List of Figures

B Preface ..... B-01
B. 1 Design process ..... B-04
C Introduction ..... C-01
C. 1 Hand made tactile graphics ..... C-06
C. 2 Computer aided graphics ..... C-07
C. 3 Computer aided graphic textures ..... C-07
C. 4 Professional tactile graphics ..... C-09
C. 5 Firebird ..... C-10
D Segmentation ..... D-01
D. 1 Level set example ..... D-11
D. 2 Level set example ..... D-12
D. 3 Level set flow diagram ..... D-12
D. 4 Level set image distance ..... D-14
D. 5 Level set example phases ..... D-19
D. 6 Label decomposition ..... D-24
D. 7 Graphic region labels ..... D-25
D. 8 PR Berkeley ..... D-30
D. 9 F score topology ..... D-31
D. 10 PR Professional and novice ..... D-32
D. 11 Professional and novice segmentation ..... D-33
D. 12 Consistency over time ..... D-33
D. 13 Segmentation Optimization ..... D-36
D. 14 PR initial algorithms ..... D-42
D. 15 PR points ..... D-44
D. 16 PR curves ..... D-44
D. 17 PR points + ..... D-45
D. 18 Trial blocks ..... D-58
D. 19 Segmentation shapes ..... D-62
D. 20 Segmentation categories ..... D-62
D. 21 Segmentation names ..... D-63
D. 22 User commented images ..... D-64
E Simplification ..... E-01
E. 1 Simplification Example ..... E-01
E. 2 Simplification Flow ..... E-04
E. 3 Removing Regions ..... E-06
E. 4 Simplification Cues groups ..... E-10
E. 5 Simplification Cue Scale Flow ..... E-12
E. 6 Simplification Outlier Removal Flow ..... E-15
E. 7 Simplification edge model flow ..... E-18
E. 8 Edge Examples ..... E-19
E. 9 Simplification Thinning Flow ..... E-20
E. 10 Thinning Examples ..... E-21
E. 11 Simplification Region Model Flow ..... E-22
E. 12 Thinning Examples ..... E-23
E. 13 Simplification Region Rejection Flow ..... E-24
E. 14 Thinning Examples ..... E-24
E. 15 Simplification Accenting Flow ..... E-25
E. 16 Accenting Examples ..... E-25
E. 17 Simplification Texture Model Flow ..... E-26
E. 18 Accenting Examples ..... E-27
E. 19 Simplification Texture Application ..... E-28
E. 20 Simplification Examples ..... E-28
E. 21 Simplification shape ..... E-33
E. 22 Simplification shape weight ..... E-34
E. 23 Simplification category ..... E-34
E. 24 Simplification name ..... E-35
E. 25 Simplification shape ..... E-43
E. 26 Simplification shape ..... E-44
F Postface ..... F-01

## List of Tables

B Preface ..... B-01
C Introduction ..... C-01
C. 1 Bandwidth of three senses ..... C-05
D Segmentation ..... D-01
D. 1 Professional consistency ..... D-32
D. 2 Experimental constraints ..... D-40
D. 3 Parameter scores ..... D-43
D. 4 Metric scores ..... D-49
D. 5 Algorithm scores ..... D-52
D. 6 User testing ..... D-61
D. 7 User testing significance ..... D-63
E Simplification ..... E-01
E. 1 Simplification cue preparations ..... E-15
E. 2 User testing ..... E-33
E. 3 User testing significance ..... E-35
E. 4 User testing user experience ..... E-36
E. 5 User testing ..... E-43
E. 6 User testing significance ..... E-43
E. 7 User testing user experience ..... E-44
E. 8 User testing significance ..... E-44
F Postface ..... F-01

A-14

## List of Terms

## Tactile Graphics

BVI Blind or Visually Impaired
Expert Person who is BVI that has expert experience using and interpreting tactile graphics
Professional TVI that professionally makes tactile graphics
Texture Repeated tactually perceptual patterns
TVI Teacher of people who are Visually Impaired

## Segmentation

Cluster Group of data points with similar properties
Contour Continuous closed edge representing the boundary of an area or region
Edge Separation between parts of an image often represented with high energy
Energy Change in information over space, often referring to color or texture gradient in an image
Graph Data structure represented by nodes connected by conceptual edges
Region Area of image separated from other parts by a continuous closed boundary or contour
Segment Single region, group of contiguous or non-contiguous regions of an image together representing similar properties or parts of the global image
Texture Repeated patterns of energy or repeated tactually perceptual patterns

## Simplification

Accented Include ancillary lines with different weight
Ancillary Line Additional lines not present for segmentation but required to understand image
Target Known dependent variable values used in supervised machine learning with independent variable cues to predict unknown dependent variable values

A-16

## Appendix Contents

G Presentation Slides ..... G-01G-29
H Experiments ..... H-01
H. 1 Segmentation .....  $\mathrm{H}-01$
H.1.1 Parameter Tuning ..... H-01
H. 2 Simplification ..... H-25
H.2.1 Regression Selection ..... H-25
H.2.2 Edge Anticipation ..... H-31
H.2.3 Region Suppression ..... H-36
H.2.4 Segment Texturing ..... H-42
. $\mathrm{H}-48$
I Proposal ..... I-01
I. 1 Introduction ..... I-01
I.1.1 Background ..... I-03
I.1.2 Manual ..... I-04
I.1.3 Automatic ..... I-05
I. 2 Project Focus ..... I-06
I. 3 Preliminary Work ..... I-07
I.3.1 Experiment Performed ..... I-09
I.3.2 Initial Results and Discussion ..... I-10
I.3.3 Initial Conclusions ..... I-12
I. 4 Project Plan ..... I-13
I.4.1 Design Process ..... I-13
I.4.2 Segmentation ..... I-14
I.4.3 Simplification ..... I-18
I.4.4 Perspective ..... I-21
I. 5 Budget ..... I-23
I. 6 Stats Base ..... I-26
I. 7 Presentation Slides ..... I-28I-47
J Literature Review ..... J-01
J. 1 Features ..... J-01
J.1.1 Color Space ..... J-01
J.1.2 Basic Operators ..... J-05
J.1.3 Basic Quantifiers ..... J-07
J.1.4 First Order Local Energy (Color Gradient) ..... J-16
J.1.5 Second Order Local Energy ..... J-23
J.1.6 Complex Local Energy ..... J-25
J.1.7 Texture ..... J-36
J.1.8 Texture Gradient ..... J-37
J.1.9 Texture \& Noise Removal ..... J-39
J.1.10 Spectral Analysis ..... J-46
J.1.11 Probability of Boundary ..... J-46
J. 2 Segmentation ..... J-46
J.2.1 Edge Detection ..... J-48
J.2.2 Corner Detect ..... J-51
J.2.3 Clustering Methods ..... J-54
J.2.4 Graphing Methods ..... J-59
J.2.5 Contour Detection ..... J-60
J.2.6 Differential Equations ..... J-60
J.2.7 Combined Methods ..... J-62
J. 3 Simplification ..... J-64
J.3.1 General ..... J-64
J.3.2 Machine Learning ..... J-64
J. 4 Assessment ..... J-68
J.4.1 User Testing ..... J-68
J.4.2 Segmentation ..... J-68
J.4.3 Simplification ..... J-80
J-80
K Related Works ..... K-01
K. 1 Publications/Workshops ..... K-01
K. 2 Concurrent ..... K-02
K. 3 Past ..... K-05
K-06
L Image Set ..... L-01
L. 1 Training Images ..... L-01
L. 2 Testing Images ..... L-09
L.2.1 Image Table ..... L-27
L.2.2 Texture Samples ..... L-30
L-31
M Code ..... M-01
.M-01
N Acknowledgments ..... $\mathrm{N}-01$
.N-01
O Vita ..... 0-01
.O-04
P Bibliography ..... P-01
P-25

## Chapter B: Preface

## B. 1 Abstract

Graphical information has become a critical method for portraying information for education, work and personal tasks and decisions. Unfortunately there are currently limited means of providing this information to individuals who are blind or visually impaired: alternate text is frequently missing, and accessible tactile diagrams tend to be time consuming to make and require expertise in order for them to be interpretable (which may be costly to the user and/or impossible to get). The aim of this project is to provide an accessible system to automatically generate tactile graphics for those who need to interpret information contained in visual images. Previous automatic conversion methods have not been especially successful and are not used in normal practice, possibly because they have not taken advantage of current advances in the field of image processing. In the preliminary work, we systematically look at the myriad of image segmentation methods that exist as part of the conversion process. For those techniques, previous researchers have often compared the results to the "gold standard" of human segmentation to evaluate their success. However, there are important difference between this "gold standard" and what is needed for tactile graphics.

Key steps by professionals who create tactile diagrams are simplification so that the information is manageable to extract through the tactile sense, elimination of perspective as it is difficult to interpret tactually, and possible spreading of information across multiple diagrams. Planned work is to examine more closely the underlying themes to the myriad of algorithm are relevant for tactile diagrams. Future work, will also involve taking the initially segmented image, simplifying it further by removing "unimportant" detail so that it is manageable by the tactile system and removing perspective based on geometric information found in the image.

## B. 2 Project Overview

## Objectives

1. To explore/modify/develop a variety of automated techniques to segment an image/drawing into a set of non-fragmented lines/curves that are effective for tactile information processing. Effective will be defined as the ability to convey information about the shape(s) of objects and their parts and/or the identification of objects and their parts, as well as identification of the main spatial relationships between objects and/or object parts in the graphic.
2. To quantitatively compare, for users who are blind or visually impaired and determining by touch alone (those with residual vision will be blindfolded), the effectiveness and ease of use of the two best automated visual-to-tactile conversion methods developed to visual-to-tactile conversions made "by hand" by expert tactile graphic makers. Effectiveness is defined as above and will be assessed by asking questions to the user about the diagram. Ease of use will be defined by the use of the System Usability Scale.
3. To explore/modify/develop a variety of automated techniques to simplify an image/drawing into a set of non-fragmented lines/curves that are effective for tactile information processing. Effective will be defined as the ability to convey information about the shape(s) of objects and their parts and/or the identification of objects and their parts, as well as identification of the main spatial relationships between objects and/or object parts in the graphic.
4. To quantitatively compare, for users who are blind or visually impaired and determining by touch alone (those with residual vision will be blindfolded), the effectiveness and ease of use of the two best automated visual-to-tactile conversion methods developed to visual-to-tactile conversions made "by hand" by expert tactile graphic makers. Effectiveness is defined as above and will be assessed by asking questions to the user about the diagram. Ease of use will be defined by the use of the System Usability Scale.

Problem Images and diagrams are becoming the sole means for conveying information: approximately $70 \%$ of textbooks convey information solely in diagram form with no text explanations [1]. To use these diagrams people who are blind require sighted individuals to convert visual images to tactile diagrams. The diagram conversion process is quite costly and can take hours to complete, that is to say, when a professional is lucky
enough to be available.

Goal This research focuses on the development of an algorithm that can automatically convert visual diagrams and photographs of objects and object scence into tactile diagrams for people who are blind without any intervention by people who are sighted. This will allow individuals who are blind or visually impaired true independence in access tactile diagrams. The focus is on providing information that is difficult to obtain by text descriptions such as shape of objects and their part, and spatial relationships between them. The algorithm will need to work with images found in everyday life such as text book diagrams and store catalogs.

Hypothesis The image processing algorithm designed here to generate tactile diagrams from input images will show statistical non-inferiority to images created by professional diagram makers when assessed in terms of their effectiveness in shape interpretation/identification of objects and their basic subparts, and of relationships between objects.

## Deliverables

1. Image processing algorithm that generates tactile diagrams from input images with similar user effectiveness and ease of use to using images generated by professional diagram makers.
2. Quantitative results assessing their performance to questions involving shape interpretation/identification of objects and their basic subparts, and of relationships between objects.
3. Set of standard training and testing tactile diagrams for future research.

## Design Path



Figure B.1: Flow of information in the design process. The process is informed by professional experience and tested with algorithm results against similar diagrams made by professionals.

## Chapter C: Introduction

## C. 1 Problem

About 285 million people worldwide are blind or visually impaired (BVI) [2]. These individuals often rely heavily on alternatives to visual information for their daily lives. Within the 2012 US population of people who are BVI only about 12\% have a bachelors degree or higher, about $38 \%$ are employed, and about $32 \%$ live below the poverty line [3]. This means less costly methods for developing visual alternatives must be explored. While the written word is relatively easy to convert and print to braille using computers, graphical information is much more difficult to supply the user with because it requires the intervention of a sighted professional. Though the visual information such as graphs, pictures, diagrams, etc. could simply be described in words, only about $9 \%$ of people who are BVI are able to read braille [3]. The individuals who create the descriptions may not always interpret the information as appropriately and prevents the user from learning as much because they are not forced to make their own conclusions. This is especially important for school children as they need to be able to comprehend and retain class material. While many schools have access to a teacher for people who are blind or visually impaired (TVI) who can assist with converting visual information to tactile diagrams and written descriptions, this service is limited because of the time it takes to make the graphics. Most adults do not have access to such costly support in their daily lives at work or at home often hindering their abilities to complete assigned tasks. The severe lack of visual information accessible by people who are BVI may also explain why fewer students who are BVI are attracted to science, technology, engineering, and math (STEM) fields, as these fields heavily involve graphical representations.

Currently, for an individual who is blind or visually impaired to access a tactile diagram, they must rely on a sighted, professional diagram maker to convert the visual diagram to
tactile diagrams. A wide variety of methods/materials can be used including: drawing boards, buildup displays, embossed paper, vacuum forming, micro-capsule paper, silk screening, etc [4]. Regardless of method, the conversion process is time consuming and often costly. The translating professional generally requires training to ensure they follow specific design principals such as: accounting for proportion/scale, perspective, inclusion/exclusion, simplification, etc [4]. The choices made during the design are often guided by specific information including: what are the necessary components, who uses the information conveyed by the diagram, how will the diagram be used, what materials should be used, what materials are available, etc [4].

## C. 2 Motivation

When tactile graphics are constructed manually by these professionals, they can take considerable amounts of time. From self-reports by some professionals, a single diagram can take from one to several hours to complete. This has decreased some with the use of regular computer aided graphic manipulation software (such as Adobe Photoshop or Gimp coupled with instructions and texture sets developed in [5]), however the time taken is still relatively long. The long development time causes there to be generally fewer graphics available to tactile readers and unless the diagrams are used repeatedly, such as in class, by different users over the years, they are generally not available immediately upon request. School areas that require large amounts of diagrams such as STEM are often not provided adequately due to the lack of feasibility. For adults, it is even worse, both in work and in everyday living environments, as they do not have the full time support that children K-12 do (at least in our state of Virginia). In addition these graphics are often cost prohibitive because many adults with visual impairments live below the poverty line.

Automated graphic conversion application would greatly decrease processing and development time while increasing the availability of tactile graphics for all ages. In addi-
tion, such an application could still be used as a starting place for further diagrammatic changes if a teacher wants to have more control over the resulting image. Some attempts have been made by researchers and private companies but they all need more work. Firebird Graphics Editor by Enabling Technologies, TactileView by Irie AT, and QuickTac by Duxbury Systems are commercially available applications for helping TVIs make computer generated tactile graphics; however, they are each limited by lack of functionality and compatibility which has lead to them not receiving widespread use. Ladner et al. looked at converting mathematical plots/graphs into tactile graphics but not diagrams and natural images [6]. Barner et al. looked into the foundation of automatic tactile graphics creation with image processing algorithms that were available at the time (1997 to 2007) but Lacked stronger modern algorithms combined with image simplification and perspective correction as well as user testing and comparison with professional work [7-11].

Though image segmentation is at the core of the conversion process because it gathers like information together and defines initial boundaries, it is not the most important. The tactile system transmits significantly less information than the visual system. This means that extraneous or unimportant information in the diagram can add noise to the tactile system, impairing the user's ability to interpret the important parts of the diagram. Many of the more accurate segmentation algorithms begin the simplification process by removing noise (additional small regions/boundaries) but further simplification is required. Besides removing information acting as noise, perspective/depth information needs to be removed. The end user will often perceive perspective lines as a change in shape or structure as opposed to depth information. When a professional designs a tactile graphic, they reduce an image to its core components and remove perspective by either reshaping the object or splitting the diagram into multiple images for each face of the object.

Further simplification will also be needed to remove clutter that might overwhelm the user with too much relevant information. This will likely be achieved through implemen-
tation of a hierarchical segmentation scheme that will allow for dynamic zooming as well as the addition and subtraction of information presented. Part of the simplification will be ensuring that any line that should be continuous remain so in the end product as a tactile user cannot perceive imaginary line completion or subjective boundaries in the same way a visual user can. This will also be where region textures are added to provide continuity for the user and help the user distinguish regions/objects. Once the algorithm has been shown to work correctly for a wide range of situations and applications, it will be combined with tactile computer peripherals to offer real time access such as during web surfing or data manipulation

## C. 3 Core Process

Regardless if the diagram author is a person or computer, the primary steps in image conversion are to segment the image into continuous regions (i.e., object parts) and then simplify the image by removing extraneous information and viewing perspective. The initial segmentation must attempt to find the most important parts of the image that need to be conveyed to the user in the same manner the human visual system would focus on important aspects of the image. Next, the system must attempt to locate and remove any extraneous information. Such information could simply be noise that has carried through or even other large objects that add too much to the diagram to be understood. Due to the orders of magnitude lower temporal bandwidth that the tactile system has available compared to the visual system, any extra information poses a comprehension threat to the user interpreting the data; this bandwidth discrepancy can be seen in Table C. 1 [7, 12]. Even though the temporal bandwidth of the auditory system is better than the tactile system it falls short of being an optimal candidate for conveying information because of its low spatial bandwidth [13]. The tactile system is also limited by primarily processing information serially as compared to the visual system's parallel processing;
this causes a dramatic increase in time spent interpreting tactile diagrams. Another important consideration in tactile diagrams is the removal of perspective. In addition, there is greater limitation of the top-down processing that is used to disambiguate objects. An image such as a building fading into the distance is generally understood by the tactile reader as a change in size or shape not as the intended perspective. At this point it is not clear in which phase of the automatic conversion process perspective should be removed because there is no current research on the topic but it is clear that perspective needs to be accounted for.

Table C.1: Bandwidth of three senses $[7,12]$

## Sense Modality <br> Limit bits/sec

| Skin | $10^{2}$ |
| :--- | :--- |
| Ear | $10^{4}$ |
| Eye | $10^{6}$ |

## C. 4 Manual Implementation

Many books, web tutorials, and teaching courses exist to help train teachers for individuals who are visually impaired (TVIs) how to develop tactile graphics [4,5,14-18]. These sources teach methods ranging in their development methods from the use of craft supplies, to image manipulation software; some of the hand- made graphics can be viewed in Figure I.1.1. More teachers are looking towards technology in order to produce consistent results and to reduce the time it takes to make tactile graphics. Figure C. 2 shows two of the most common types of computer aided tactile graphics printing methods. The first shows micro capsule paper which works by small capsules expanding due to heat resulting in raised lines or textures. The second is a braille embosser which works by creating small bumps in paper in an attempt to resemble lines and textures. Figure C. 3 shows textures that were determined to be salient and distinguishable for use in tactical graphics and line drawings. The use of texture is important in helping the tactile reader
mentally separate different regains or group like regains together that would otherwise have no distinguishable relation $[4,6,18]$.


Figure C.1: Example hand made tactile graphics $[4,19])$


Figure C.2: Example computer aided tactile graphics machines


Figure C.3: Example computer aided tactile graphics textures (not to scale) [18]

## C.4.1 Modeling

This project began with observing professionals converting graphics to tactile diagrams in order to model an automated system after their work such as the images in Figure C.4. The observed procedure is similar to previous observations by other researchers [6]. Working with the professionals led to a procedure for using image manipulation software in order to develop tactile graphics with textures [5]. The professionals, using this procedure, converted a set of images to tactile graphics allowing for a basis of comparison for further work. Though the resulting images often look similar there are naturally small visual differences that may be perceived tactically as wide variations. These differences result in the "human" point in the validation metrics discussed later.

Professional work flow based on observations by Ladner [6] and modified to fit recent protocols for using Image manipulation software [5]:

1. Planning
2. Scanning (if needed)
3. Perspective Correction
4. Image Outline \& Simplification
5. Applying Texture
6. Add Braille Text
7. Add Key/Legend
8. Proofing and Touch up
9. Printing


Figure C.4: Example tactile graphics from professionals (not to scale).

## C. 5 Automatic Implementation

Current commercially available software attempts to allow users to develop their own tactile graphics but often requires that the image author make the graphs manually. The top 3 commercially available applications include: Firebird Graphics Editor by Enabling Technologies, TactileView by Irie AT, and QuickTac by Duxbury Systems. Each of these allow the user to manually draw and edit diagrams and to export to a braille embosser. Only TactileView allows the user to export to micro capsule paper however it is limited to importing vector images which are specialized to diagrams and not intended for natural images. Only Firebird Graphics Editor allows arbitrary images to be imported while attempting to convert the images automatically. Firebird Graphics Editor by default uses a 2 segment clustering method to convert the graphic but also supports edge detection. When the software separates foreground and background it arbitrarily adds its default texture to one of them but does not leave an outline between regions (as recommended by TVIs). FireBird Graphics Editor is also the only editor that allows the image author
to use predefined textures to help distinguish regions. I.1.2 shows examples of FireBird Graphics Editor's automatic image outputs. None of the commercially available application do an adequate job of converting the graphics for a wide range of uses nor do they have enough options or features for most uses, such as: import arbitrary images, outline important components, multiple textures for regions of interest, and the inclusion of ancillary lines to accentuate structures of interest. They also do not attempt to simplify the diagram or remove perspective.


Figure C.5: Firebird Graphics Editor in action

For several years researchers (Ladner et al. \& Barner et al.) have looked at the process of converting visual images to tactile graphics and attempted to devise an automatic conversion algorithm [6-11, 20, 21]. In 1997 Barner developed a program to automatically convert images [7, 9]. Unfortunately, while a great foundation, development of this application stopped in 1998 and is no longer compatible with current operating systems. Documentation for this software indicates that it used sobel edge detection and kMean clustering based solely in red/greed/blue (RGB) color space. The comparison in this research and later attempts to use watershed based segmentation [11] are limited to al-
gorithms that were available at the time and lacks stronger modern techniques. They also appear to lack identification comparison to diagrams made by professionals. Barner has also worked with automatic conversion of vector images focusing on edge detection [8]. This was at a time when raster images tended to be small with poor resolution and vector images were thought to become more common due to smaller storage requirements. Since then space has become less of a problem and better loss-less compression algorithms have became more popular resulting in the decline of vector images. Because vector images do not lose information when they are scaled, they perform better with diagram type images; however, they are not equipped to handle real life photographs. Though they performed user validation testing, based on the information available in their publications, it seems as though they never compared the user's ability to identify their processed images with images made by professionals $[8,9]$.

In 2005 Ladner worked on an automatic image conversion application modeled after professional work flow focusing on science, technology, engineering and math (STEM) text book graphs [6]. This early work was based on using segmentation and optical character recognition (OCR) to find and temporarily remove text, simplify, and add texture, followed by adding the text back in braille form. In 2007 Ladner in collaboration with others developed a software assistant that works with an OCR program and Adobe Photoshop or Illustrator to help convert STEM graphs [10]. This work is limited to mathematical plots/graphs and is not intended for use with natural images. Based on the information available in their publications it seems as though they never performed user validation testing.

Recent work, in 2014, also mentioned the process of converting images to haptic form but their work focused on the implementation of a display device [22]. They accepted an already implemented segmentation algorithm based on graphing [23]. They mention that that algorithm is greedy and not intended for real time as their work focuses on the display of haptic information. Their user testing focused on inclusion/exclusion of texture
and inclusion/exclusion of multiple levels of detail with no comparison to images created by professionals.

## Chapter D: Segmentation

## D. 1 Introduction

Segmentation in the process of splitting an image into its constituent components or segments. A segment does not necessarily have to be a contiguous area it can be broken into closed bounded regions separated by some distance. Segments are one or more regions that schematically relate between and within the area of the region(s) in some context. For example the American flag may be split so that each star and each stripe are split into individual regions but all the red strips are grouped into one segment, all the white stripes are grouped into one segment, and all the stars are one segment.

Though past researchers considered edge detection and segmentation this was revisited to determine if newer methods could be applied. One focus of this project was to model the algorithm after the way professionals see and process tactile diagrams. To that end a segmentation model is needed to account for the initial steps taken by professional diagram makers.

## D. 2 Design Decisions

There are a wide variety of existing image segmentation algorithms designed for other applications such as object tracking, industrial manufacturing inspection, microscopic cell labeling, and medical imaging. All of these algorithms use a combination of the degree of similarity between pixels and/or more feature dimensions to determine the segmentation. The different algorithm classes considered here represent significantly different approaches to segmentation: edge detection, clustering, graphs, contour detection and differential equations. Of these classes, only edge detection and clustering have been previously examined for this automatic creation of tactile graphics.

## D.2.1 Features

Images features are any low level easily extracted pieces of information about an image including: color, energy, and texture. Color space is a way of storing the color data for each pixel. Gray scale and red/green/blue (RGB) are often known as the ways TV and camera systems work. Cyan/Yellow/Magenta/Black (CYMK) is how color printers encode information because they are depositing layers of pigment. Hue/saturation/level (HSL) is often seen as the honeycomb color picker in many word processors and other desktop applications. Often a measure of distance between two pixels' colors is needed to determine how different they are. Energy can be seen as slightly higher level information because it requires a change in color. The rate of change in color represents the energy magnitude in any local area of an image. Energy not only has a magnitude but also has an orientation (direction of change) and phase (shift in frequency components). Texture (patterns of energy) can also be extracted from an image in 2 ways as a gradient (change from low-high textured area) or as a classified texture (giving each texture occurrence a label). A more extensive view of the features described below including calculations can be found in Appendix J. 1 .

Color Space Logically it makes sense to model as much of the system as possible after human vision and perception since it effectively segments images into regions easily understandable by human cognition. One area that has been particularly considered in this regard are the color spaces. Gray and red/green/blue (RGB) are obvious options to consider for representing color because of their ubiquity in many types of media. However, they are not necessarily based on human perception. Lab/Luv color spaces (brightness, red-green, blue-yellow) were originally designed to best fit human perception with each increment matching to the same unit step in human visual perception [24]. Throughout the thesis work four color spaces were explored: gray/RGB for their ubiquity and Luv/Lab for their base in human perception in an effort to model segmentation after the human
visual system.

Distance Metrics Various distance metrics can be considered both in image space and in feature space when the feature is found on a continuum. The most commonly used distance metric is simple euclidean distance; it is used primary because of its simplicity and speed. The next most commonly used metric is normalized Euclidean distance. In this case the distance is normalized by the standard deviation of all points in a group/cluster. The normalization becomes useful when comparing distance within and between clusters of data points where each cluster has its own standard deviation. Mahalanobis distance is the next logical progression because it not only accounts for standard deviation in general but the deviation and correlation between different dimensions [25]. Mahalanobis distance does not provide much added benefit when variances are small because of the extra calculation time needed especially when it needs to be run many times [26]. Several other less commonly used distance metrics exist including: city block (taxi cab), minkowski, chebychev, cosine, spearman, hamming, and jaccard. Some correlation metrics are also referred to as distance metrics such as: chi squared distance and earth movers distance, but these are generally intended to quantify the difference in histograms.

Energy is a measure of how different spatially adjacent pixels are. At its most basic this builds on the concept of using distance metrics in color space to form a derivative through convolution kernels such as central difference, prewitt, and sobel. These kernels can also be made directional such as with the kirsch operator, or even include the second derivative with the laplacian operator. All of the energy operators work to find areas of rapid change between the pixels usually relating to their color. Some energy operations become even more complex in an attempt to capture many of the nuances related to finding changes between pixels such as the monogenic signal, phase congruence, SUSAN filter, etc. In summary, many of the energy operations not only give a measure of
pixel change but also the orientation or direction in which the change happens and the predominant phase among frequency components.

Texture in an image is representative of repeated patterns of energy. This can simply be extracted from an image using measures of how textured a region is (entropy, third moment) or it can be extracted using classifiers (textons) to represent which texture is present in a given area. Local entropy is the simplest method for determining how textured a given region is, as a representation of the chaos. The third moment is also relatively simple to extract and represents the amount of repetition in a local area. [27,28]

Human Equivalent The human visual system is theorized to not only convert the color systems but also split frequency bands. Within these bands the human visual system then searches for edges at different directions (energy) or repeated patterns (texture) $[29,30]$. This is similar to computational image processing algorithms: filtering, local energy, oriented energies, edge detection, and texture extraction. An aim of this project is to model the algorithm on how a professional diagram maker would approach the situation consciously and innately. Therefore the algorithm must account for human visual perception that drives the way they would segment images.

## D.2.2 Classes

Building on previous work in visual to tactile image conversion [8-11], this research seeks to further consider new automatic image conversion and segmentation methods developed in the computational vision field not previously applied to visual to tactile image conversion. Several classes exist for image segmentation such as contour detection, graphical representations, clustering and even differential equation based approaches [31-36].We explored the algorithm space of these methods to determine the best algorithm for our application to ensure similarity to segmentation performed by TVIs and speed.

Edge Detection Edge detection is the process of examining image energy to find areas of rapid change in color or texture indicating the occurrence of an edge. Basic edge detection was not considered as a core algorithm because it is too susceptible to noise and it does not produce continuous contours for region boundaries that can later be filled with tactile textures. Edge detection could still, in subsequent steps of visual to tactile diagram conversion, be useful for adding ancillary information back to images after regions are found. However, we will not consider it for the segmentation step. For details on the several edge detection methods already developed see Appendix J.2.1.

Clustering Clustering is the process of grouping like pixels with each other to signify regions. These algorithms require data to be represented on a continuum, such as colors from dark to light or textures from low to high textured areas. There are a variety of different clustering algorithms. Two core clustering algorithms were examined, for visual to tactile diagram conversion, namely Mean Shift [32] (grouping pixels based on a density gradient in a feature space) and k-means [37] (grouping pixels based on distance to a cluster center in feature space). Previous work normally considered these algorithms solely in color space, as in the two aforementioned papers; although others have included texture [38]. Though mean shift does not require knowing the exact number of segments to be formed before running (as with k-means), it was quickly pushed aside because it tends to be computationally expensive and results in an over-segmented image. Kmeans requires knowing the exact number of segments before running but tends to run rapidly compared to other algorithms and still produces viable results. K-means requires that several parameters need to be set before it is used, specifically: the color space, the inclusion of other dimensions such as texture, the number of clusters, the initialization type, and the distance measure. K-mean's parameter space is further explored in Section D.5.

Graphing Graphing is the process of segregating pixels into groups based on dissimilarities. Graphing based methods $[34,35,39]$ gained popularity for a time because, unlike clustering methods, they easily incorporated more low level feature information that did not necessary have to be on a continuum. This is achieved by treating pixels as nodes in a graph with edges between neighboring nodes. The algorithm then progressively cuts edges between dissimilar nodes until it has split the graph into smaller sections or segments. Unfortunately, though the algorithms may produce usable results, they tend to be computationally expensive, taking more than 10 minutes, and more often up to hours for modern image sizes, to run on a home computer [39, 40]. This would be problematic for out envisioned goal of having real-time conversion of visual to tactile diagrams for dy namic tactile computer interfaces. We therefore, will not consider graphing methods for out applications.

Contour Detection Contour detection is the attempt at locating parts of an image that represent a boundary between regions. This is similar to edge detection but requires closed contours as an output. The closed contoured regions can then be grouped together based on similar color to create segments that can be textured (an essential component of tactile diagrams). The current dominating contour detection algorithm uses the global probability of a boundary with an oriented watershed transform (gPb-owt) [41]. This algorithm allows for the incorporation of several features via linear combination, specifically multi-scale gradients and spectral information that help to find edges and boundaries between regions. Watershed transforms use gradient information as a topology showing where water would flow to if dropped at any point in order to form closed boundaries from edges. For the gPb -owt algorithm, they are used similar to previous work by Hernandez in image segmentation for the intent of making tactile graphics [42]. Arbelaez's work however, adds some extra fine-tuning with specific gradient filters and the addition of an orientation parameter to remove noise and over-segmentation. The gPb-owt algorithm has already been optimized for general image segmentation using

Lab color space, component weights, texture classification, and a range of smoothing filter with specific sizes and shapes. gPb-owt-ucm expands the method further by adding a single parameter that determines the amount of detail that is included.

Differential Equations Differential equation based segmentation works on the principal of an initial contour propagating over an image to split regions based on what is inside versus outside the contour. These algorithms often have complicated differential equations associated with them that need to be solved to determine how the contour will propagate. The concepts behind graphing methods (comparing nearby pixels) and active contours (propagating contours) come together with the idea of embedding a problem in higher dimensional space to create the level set method $[36,43]$. This method moves each pixel up/down past a zero set plane in a third dimension based on a feature set. In the cited papers, the feature set was pixel color, image space proximity to an intense gradient, and contour stiffness, with the zero plane crossing producing the contour line. Any feature can simply be added to the calculation by determining whether or not the pixel should move up or down in the new space based on the feature's values. This algorithm has become popular in medical image processing but has also shown promise with natural images and diagrams. The major problem with this algorithm is the vast number of parameters that need to be set: color space, inclusion/exclusion of texture, weighting of components, number of phases, color space distance measure, image space distance measure, gradient type, and initialization type.

## D. 3 Design Implementation

While the base algorithms used already existed, several additions/modifications were incorporated to potentially improve performance for K-means and level set algorithms. The additions were later tested with a wide variety of user inputs to find the best parameter set for visual to tactile diagram conversion (Section D.5).

## D.3.1 K-Means

K-Means works predominantly by iteratively updating cluster means by regrouping points in feature space by their distance to the closest current cluster mean. K-means allows many features to be used if they represent a difference or similarity that pixels can be grouped from such as color. However, features that represent a boundary between clusters, are uniformly distributed, or labels, such as gradient, image space location, or textons cannot be used. Though more features can always be added this has diminishing returns and introduces noise, therefore measured used within the algorithm was the target for improvement. The effectiveness of current color distance metrics in segmenting an image are limited by assuming variance in each dimension are equal and assuming change in brightness is equally important as change in color. As such, the research focus was on improving k-means by considering color distance metrics that address these limitations. Initially, many standard metrics that are widely used were explored, such as euclidean distance, city block, cosine, and hamming distance. As Euclidean distance produced the most effective segmentation for visual to tactile diagram conversion in preliminary testing, it was used as the starting point for developing new metrics and the others were not tested further.

When considering images that have a color tint throughout, brightness/dullness throughout, or only small variation within a particular color dimension we noticed Euclidean distance did not perform as well. The first alternative considered was based on the observation that each cluster may have different standard deviations that might help determine if a point should belong to it or not: Theoretically a broader cluster should be more likely to accept a distant point than a tighter cluster. This led to to a distance metric in which the Euclidean distance to the closest current cluster sphere center was normalized by the cluster's standard deviation independently for each dimension (Eq. (J.1.5)). The second alternative considered was to maintain the idea of variation in tolerance of acceptance based on variations in standard deviation but allow it to be difference not only
in each feature dimension but also have an orientation to it. This was achieved using the Mahalanobis distance which standardizes the deviations along principle components (Eq. (J.1.6)).

Another possible issues when considering a color distance metric is the problem of varying light sources and shadows in the scene. For example, a person standing beside a tree may have part of their face and cloths in sunlight and part in shadow of the tree. In terms of RGB color space, the pixel values between the face/jacket in sunlight and in shadow vary significantly. However, the perceived color (and the part of the object) remains the same.

We then sought to determine if points that are collinear with respect to the origin might likely be better grouped together. In terms of RGB color space this means points that have roughly the same color but different brightnesses may still belong together. The angle between two points with respect to the origin is the arc angle or dihedral angle (Eq. (J.1.7)). Combinations of dihedral angle and each of euclidean distance and normalized euclidean distance were implemented. To do this only the first 3 dimensions of the dihedral angle were considered in cases where texture added higher dimensions. Traditionally only max iterations are considered with k-mean but we also created a max time parameter because thousands of operations were going to be tested and needed to be run in a timely manor. The relationship between the pixel of the same object in light and shadow can be captured mathematically by assuming that pixel location that are collinear with respect to the origin might be better grouped together. The degree to which this occurs can be determined by the arc angle (dihedral angle), which is the angle between tow points with respect to the origin (Eq. (J.1.7)).

Based on the above considerations, Mahalanobis distance, Euclidean distance, and dihedral angle were each tested alone, as well as, combinations of dihedral angle and each of euclidean distance and normalized euclidean distance were implemented. To do this only the first 3 dimensions of the dihedral angle were considered in cases where
texture added higher dimensions. In addition, traditionally the k-means algorithm run-time is limited by the maximum iterations that can occur. However, as time taken is significant concern in allowing real-time interaction with media (such as on the internet), a maximum time parameter was also assigned. Normally k-means requires the number of clusters to be known before run-time but each tactile graphic often has a different number of segments depending on image complexity and content. To allow for a dynamic cluster count we implemented a way for $k$-means to try several cluster counts in a row then select the segmentation results based on the inter cluster distance (Appendix J.2.3) [44].

## D.3.2 Level Set

The level set method is a way of taking an image segmentation problem as a propagating boundary $(\Psi)$ in differential equation form (Eq. (J.2.14)) and simplifying it to an iterative linear multiple polynomial form $\left(\phi_{n}\right)$ by embedding the problem in one higher dimensional space. It combines image features in different forms to force pixels of the extra dimensional object up or down. This is accomplished by grouping features into four sets/components: image space plus extra dimension $(\phi)$, grouping space $(\omega(\phi))$, cutting space $(\lambda)$, and damping space $(\gamma(\Psi))$. Image space consists of pixel location relative to other pixels, and the extra dimension. Image space provides the base for other spaces to influence the extra dimension upward or downward. Grouping space consists of color and color cluster distance which push the extra dimensional shape away from zero. Cutting features consist of gradients which push the extra dimensional shape towards zero. Damping features consist of contour curvatures which reduce the effects of the other components. With each additional extra dimensional layer, called a phase, the number of potential segments doubles because each segment is based on the high/low value of each phase together.

Level set alone is often used with simple images without much detail in order to find an object, such as a car in traffic camera footage or a box on a conveyer belt, even when
there is noise or poorly defined edges (Figure D.1). It is also used with simple but noisy images applied to medical research, such as some medical images of the body to locate different tissues or in images of cell cultures to locate individual cells (Figure D.2). Level set has not been abundantly used with photographs and diagrams and, when it has, was not shown to score well in past research seen latter in Figure D.8. This is because it is specifically designed to ignore highly detailed areas as it assumes the details are noise. However, highly detailed areas are not able to be perceived in tactile diagrams. Therefore, level set may not show the same limitations for our applications. We sought to explore this additional use for level set by expanding its functionality and implementing more features because of level set's ability to incorporate many types of features where other algorithms are limited to improve efficacy for tactile graphics.


Figure D.1: Level set example showing the results and propagation of 3 phases producing 6 segments [36].


Figure D.2: Level set example showing the results of a single phase on a noisy image with poorly defined edges [45]. Shows the image with the contour ( $\Psi$ ) and the level set (image space with extra dimension, $\phi$ ) with the same contour. Double click to open animation.


Figure D.3: Level set flow diagram version developed. Double click to open animation.

## D.3.2.1 Image Space

Image space represents the ( $\mathrm{x}, \mathrm{y}$ ) coordinates of the image along with the extra dimensional object as a z dimension or phase $(\phi)$. The high values of the phase represent pixels within a segment and low values represent pixels outside of the segment. The segment boundary is the zero crossing contour on the $z$ dimension of the phase known as the zero set. Two or more phases can be used where each is above and below zero. Pixels low in either phase represent segment 0, pixels high in one or the other phase but not both represent segment 1 and 2, while pixels high in both phases represents segment 4.

Within image space, the propagation of the phases are not only influenced by the other components up/down force but also by the distance between pixels from the phase's zero
set both within and outside the zero set. Ideally pixels closer to the zero set should be allowed to change between upper/lower halves more easily, whereas pixels further from the zero set should resist transitioning.

Heaviside The simplest method for marking pixels in and outside of the phases would be as purely binary. When applying the up/down force from components, an almost purely binary extra dimensional space (Heaviside function) would make any pixel just as likely or just as unlikely to move from one segment to the other [46,47]. Allowing pixels anywhere to move with the same likelihood, prevents the inclusion of pixels' proximity to each other as a feature, and allows any pixels to jump segments due to noise more easily.

Linear The second easiest method for modeling pixel distance is to mark each pixel distance from the phase boundary in a linear fashion. This method benefits from allows for pixels near the segment boundaries to transition more easily between segments as the contour propagates. However, it also makes it nearly impossible for pixels far from the contour to transition to the other segment even with tremendous force from the other components. Unfortunately, it also adds to computational complexity by having many more values compared to Heaviside.

Plateau A trivial method for correcting the weaknesses of the last method would be to plateau the linear distance before the ends are reached, creating a ceiling and floor. This allows pixels near the contour to move like linear but not preventing movement at the extremes of linear. It also keeps computational complexity lower with fewer values. Computational complexity and ultimately run-time is reduced because repeated values in the phase and other components during calculations are rapidly pulled from CPU cache.

ERF To further improve the smoothness of the transition between phases and the likelihood a pixel should transition a sigmoidal shaped function was used in the form of the error function (erf) [48]. The erf was selected for this application as it represents
the cumulative sum of a normal distribution which, by itself, should balance calculations for normally distributed likelihood data of pixel movement up/down. We implemented a parameter for later tuning that scaled the erf allowing it to exist between a linear and Heaviside function to find the best model for our image set. Figure D. 4 shows examples of each distance representation possible in image space.


Figure D.4: Various methods for describing distance from a boundary in image space ( $\phi$ ) shown as calculated from a segmented image with only a square in the middle. The black contour represents the segmentation boundary (phase zero set), with pixels above zero inside the segment and pixels below zero outside the segment.

## D.3.2.2 Grouping Space

Grouping space works similarly to the k-means algorithm in that colors and textures are clustered in grouping/color space. In this case grouping is a function of $\phi$ such that
pixels marked as being in a phase are marked as being in the same cluster in grouping space. Then each pixel's distance to each cluster's center is measured to determine which cluster the pixel should belong to in the next iteration [48]. When texture is included it must use a measurement of texture that is a continuous representation of how much or how little texture is in an area; this is similar to how the RGB color measurement is how much or how little of a color is present. At its core, distances between clusters in color space are needed to determine the likelihood a pixel belongs to each cluster. Some of the more appropriate distance metrics were explored for the developed algorithm. The use of the traditional euclidean distance was improved upon with a standardized euclidean distance (SED). The SED accounts for not only the pixel's color distance from the cluster center but also how compact the cluster is. The dihedral angle, the arc angle between the pixel and cluster center, was considered because, in RGB space, colors could be far away but represent only a change in brightness (all colors) not just a single color. A more modulated approach was also considered through a distance measure based on the combined use of the euclidean distance with the dihedral angle. This was to take into account that a change in brightness is not necessarily unimportant but it is not expected to be as important as a change in a single color. Finally, the use of the mahalanobis distance was explored to account for non-spherical clusters. Each of these were tested in later optimization phase.

The likelihood a pixel belongs to each cluster based on a distance measure then needed to be converted to a force to push the extra dimensional object either up or down, depending on which segment in more likely. Other researchers kept each phase's force calculation separate from each other. However, we wanted to allow the number of phases and dimensions why? To change based on the image, so an adaptive method was needed. Since a pixel may equally seem to not belong to the segments represented by each phase alone and also likely belonging to the segment represented by the overlap of both phases, we needed to modify the calculation of the extra dimensional force
to capture this. If left along it would attempt to push both phases down but in reality they both need to be pushed up so the pixel can belong to the segment represented by the two phases together. The extra dimensional force needed to correctly influence both phases, while also balancing the possibility of a pixel equally likely to belong to two different phases not only pushing one up but both up. This was achieved with Matlab, by adding vectorization and sign multipliers to each phase before incorporating the likelihood of belonging to each cluster. This allowed the small and large probabilities for or against inclusion to influence the extra dimensional shape. This also allowed for near ties to be kept here an settled by other components outside of grouping when all components are brought together. To ensure grouping features did not over power other components with large values the pixel forces were capped and floored with erf.

## D.3.2.3 Cutting Space

Traditionally, level set cutting is based on intensity changes or gradients. However, the gradient method must be selected and, to deal with the resulting noise, a noise reduction technique must also be selected. To reduce energy noise from dithering, a Gaussian blur was implemented before gradient calculations. Four gradient calculations were then explored: the basic central difference derivative estimation, the monogenic signal as a smoother more accurate gradient, and Berkeley's multi-scale probability of boundaries and global probability of boundaries [41]. Value ranges were altered to fit with our sum of weighted components. Texture gradients, such as susan, were not used because they were too abrupt, having almost binary representations. That type of representation would be too harsh and controlling when balancing level set components. Smoothing parameters for kernel size and sigma along with gradient weights from each color layer were later optimized along with the gradient method. Dynamic weights of each color layer based on the image's standard deviation were also implemented.

## D.3.2.4 Damping Space

Traditional damping, calculates the curvature of the segment boundary contour ( $\Psi$ ) so that noise creating high curvature or sharp points do not overly influence the end segmentation. This was altered to fit into our weighted sum as opposed to traditional multiplication.

## D.3.2.5 Weighted Sum

Each component (i.e. grouping, cutting and damping) needs to be combined to shift pixels in the extra dimensional space $(\phi)$. The way these are combined will influence how well the level set technique works for different images types. Typically the grouping components $(\omega)$ are weighted and added to weighted values of cutting the components $(\lambda)$ multiplied by the damping components $(\gamma)$ and the previous image space with extra dimension $(\phi)$ plus a small push to prevent falling in local extrema $(\varepsilon)$ [36]. Other implementations also add a speed increase (c) added to the damping components [48].

The way damping and cutting components are traditionally combined using multiplication, although prevents noise from taking over, limits critical details needed for tactile graphics. After trying several different component combinations, we decided to split cutting and damping into their own weighted sum components while still multiplying each with $\phi$. This allowed for damping effects with high noise but still allowing for added details. Such changes required altering the number range each component worked in compared to other research methods in order for each to apply force in their desired direction for each phase. Comparatively, other work uses the reciprocal of cutting $\left(\frac{1}{\lambda}\right)$ so that large gradients push $\phi$ towards 0 . We invert the normalized cutting range $(-\lambda+1)$ again so large gradients are 0 but also accounting for the overall min/max gradient in the image. Eq. (D.1) shows the two common methods plus the method developed here. To further improve the method we implemented a weight change over time allowing for different components to take control at different phases of the convergence process. Component
weights and weight changes rate were optimized in Section D.5.

$$
\begin{align*}
& \phi^{\prime}=w_{1} \cdot \omega(\phi)+w_{2} \cdot \gamma(\Psi) \cdot \lambda \cdot \phi+\varepsilon  \tag{36}\\
& \phi^{\prime}=w_{1} \cdot \omega(\phi)+w_{2} \cdot(c+\gamma(\Psi)) \cdot \lambda \cdot \phi+\varepsilon  \tag{48}\\
& \phi^{\prime}=w_{1} \cdot \omega(\phi)+w_{2} \cdot \lambda \cdot \phi+w_{3} \cdot \gamma(\Psi) \cdot \phi+\varepsilon  \tag{D.1}\\
& \text { [This work] }
\end{align*}
$$

## D.3.2.6 Initialization

Typically level set is initialized by user input. However, for automatically generated tactile graphics, initialization needs to be automated. Some researchers implement a series of circles for each phase or many circles per phase as shown in Figure D.5. We found that the exact size and placement of these circles influenced how well level set performed. We improved on the standard of selecting a single circle size and placement for all images by developing methods so that optimizable parameters were available for circle size, padding space between each circle within a phase, and circle placement shift between each phase. We went further by developing two new initialization techniques. The first was to use the output of k-means as an initial state since k-mean is already a well researched algorithm. The second was to automatically estimate the pixels representing the background using histograms. In this method, the five largest peaks that were no further than 3 bins away from the absolute largest, representing more than $10 \%$ of the image each, were placed in the background segment at the start of level set. The parameter values were selected after an initial round of testing that found these values to best select background content not important to tactile diagrams.


Figure D.5: Level set example showing the results and propagation of various initial conditions with 2-3 phases. [36].

## D.3.2.7 Clean

Two alternatives were considered to reduce noise. In the first method, small segments were removed and the corresponding pixels were labeled as background. In the second method, small regions had the potential to be absorbed into the background or neighbor regions. Each of these were tested in later optimization phase.

## D.3.2.8 Phase Count

The maximum potential number of segments or clusters is doubled for every additional phase as segments are all logical possibilities of each phase being high or low. If a logical combination is not present then the constituent segment is also not present. Testing different number of phases and ultimately different number of potential segments is essential to see which converges better to professionally made tactile graphics. In the past, the number of phases must be selected prior to run time and maintains the same number throughout. We developed a method that repeats level set segmentation with 1-3 phases
then selects the best based on inter cluster color distance (Appendix J.2.3). This metric measures how well clustered the pixels are in grouping space.

## D.3.2.9 Reduce Complexity

To reduce computational complexity and boost speed for future real time tactile graphics, periodic checks during run time clean out unnecessary calculations. Since overlapped high phases take more time to calculate, larger segments are moved to nonoverlapped phases and smaller segments are moved to overlapped phases. This is performed by simply relabeling pixels in smaller segment to belong to higher numbered segments. This also required that segment labels always corresponded to the same phases and phase overlaps. Plus higher segment labels always represented overlapped phases similar to the way most people always write truth tables as if they are counting in binary. If an entire phase was no longer needed, due to reduced segment count, then the extra phase was removed to reduce computation time. This also prevented small new segments from emerging and creating noise.

## D.3.2.10 Merge

To further reduce noise, segment color (grouping space) clusters with overly specific, small standard deviations, that reject the inclusion of similar colors needed to be avoided. This will both reduce the complexity of the resulting diagram and the chance of falling into local extrema. Two possibilities for merging groups are to merge clusters if the clusters' color standard deviation overlap or to merge clusters if the clusters' color means are close together. Each of these methods was tested in later optimization phase.

## D.3.2.11 Split

In the event a segment is made up of colors with a large standard deviation, the algorithm can force the segment cluster to split into smaller clusters. This could be needed
is a segment and its grouping cluster grow to take over too much of the images reducing details. We thus, added the ability to split segments to the algorithm. Three possible methods were considered for splitting segments: histogram peaks, k-means with only the subset in question to make 2 new segments, or k-means iteratively trying different number of new segments to find the most likely best number of new segments. Each of these were tested in later optimization phase.

## D.3.2.12 Stop

Traditionally level set it run for a predetermined number of iterations. In order to reduce computation time for future real-time applications with tactile graphics a stopping function was created to prematurely stop iterative processing. This function measured the normalized mutual information (NMI) to determine if the current and previous iteration segmentation results stayed relatively the same ( $\mathrm{NMI}\left(\operatorname{segs}_{t 1}\right.$, segs $\left._{t 2}\right)$. It checks to see if the NMI is above a threshold indicating there was little change and the segmentation is not likely to change significantly if allowed to run longer. Unfortunately, sometimes the NMI does not reach ideal values even when the contour propagation has reached steady state. This happens if there are a large number of small regions within the segments that reduce the NMI. As a backup measure, if the mutual information (MI) between iterations does not change then the algorithm stops $\left(\Delta \mathrm{MI}\left(\operatorname{segs}_{t 1}, \operatorname{segs}_{t 2}\right), \mathrm{MI}\left(\operatorname{segs}_{t 2}, \operatorname{segs}_{t 3}\right)\right.$.

## D.3.3 Texture Application

Texturing the segments of tactile diagrams significantly improves performance over the use of raised line drawings $[18,49]$. Segments were textured according to the size of the area covered. The largest area, assumed to be background, received no texture and the following 6 largest areas received preselected textures (Appendix L.2). The algorithm used the same texture set as that used by the TVIs.

## D. 4 Optimization Preparation

## D.4.1 Image selection

The images used for digital training were selected from topic areas that individuals who are blind or visually impaired encounter in everyday life and school. The images included photographs and diagrams used in teaching science, geography, history, everyday objects, places, and product advertisements. The images primarily contained a single type of object, although some were images of scenes. The images also include common image processing challenges such as: dithering, low print quality, solid color ranges, similar intensities across boundaries, single object with multiple colors, and small gradients. TVIs were involved in picking out the images based on their familiarity to their students. The image training set of 37 images can be viewed in Appendix L.1.

Image preparation The image resolution was standardized at 6 pixels per mm with each image maximized for printing on $8.5 \times 11$ inch micro-capsule paper with 0.5 inch margins. The stated resolution was chosen because it allows for each of the desired tactile textures (Appendix L.2) to be presentable.

## D.4.2 Professional Diagrams

To determine the ground truth of what the resulting tactile diagrams should look like, first a pair of TVIs were asked to separately make tactile diagrams from the images selected. They were introduced to use of Photoshop to create outlines and apply textures to regions. They both received training on the use of Photoshop to make tactile diagrams by being introduced to how Photoshop could be used to create to create outlines and apply textures to segments [5]. They were provided with a standard set of textures to use in Photoshop based on previous work and seen in Figure C. 3 reduced by TVI selection to only 6 shown in Appendix L.2. The TVIs were instructed to design graphics as they would
for their BVI students: This involved creating line diagrams with textures while selectively removing content as they normally would to make the images understandable to their students. They were encouraged to apply textures to the image in a meaningful way in accordance with their TVI training to help students better understand the diagrams. Throughout the process the professionals were encouraged to share their methods and insights to better develop a working model of the process to help design the inevitable computer algorithms.

Then, the professionally made electronic drawings needed to be converted into segments, regions, and boundaries for assessment purposes: comparison to each other, to novices, and to the computer algorithm generated diagrams. In order to do this, a laboratory research assistant converted the line drawings with textures to segment drawings with colors marking unique segments and acting as labels. Then, Matlab code was developed for converting segments to regions and boundaries. A blurring filter was also used to visually simulate the lower resolution of the tactile system, it was expected that boundaries did not need to be as precise for tactile diagrams as for visual diagrams. The blurring filter was based on the filter equation describing this lower resolution given by Loomis (Eq. (J.4.4)) [50]. The decomposition process along with assessment metrics which can be determined in each phase of decomposition are shown in Figure D.6. A segmented image under evaluation is scored against each professional then combined by either mean or maximum depending on what the metric requires. Figure I.3.1 shows an example conversion to labeled regions.





Figure D.7: Example tactile graphics from professionals converted to region labels (not to scale).

## D.4.3 Novice Diagrams

In order to get a better understanding of the decisions professional tactile diagram illustrators make and to help formulate better discussion with the professionals to understand their decisions, we looked at how novices segment images. Five novices were asked only to use Photoshop to draw outlines on the images to segment them into their most important objects/components. They were not asked to add textures. The novices only segmented the 37 training images not the later testing images (professionals illustrated both). The novice made diagrams were prepared the same as the professionally made diagrams for processing and assessment. Same decomposition was used for novices images and in ground truth image preparation (Figure D.6).

## D.4.4 Professional Rater Reliability

Before assessing performance of the computer segmentation algorithms further analysis of the human generated segmentation was needed to ensure rater reliability and thereby, meaningfulness of ground truth. Checking for reliability was also expected to help generate better discussion with professional in understanding their methods. In this experiment the professional diagrams were compared to each other for consistency and to a group of novice diagram makers as a benchmark.

## D.4.4.1 Methods

In this experiment, the professional diagrams were compared to each other for consistency and to a group of novice diagram makers. Of the many assessment methods for segmentation accuracy explored in Appendix J.4.2 only a select few are used in this section to determine professional segmentation consistency, as well as novice segmentation boundary validity: precisions-recall, F-score, distance to professionals, Loomis distance, and probabilistic rand index.

Each segmentation image being assessed is scored against each professional segmentation image then combined by averaging between the professionals then between the images. Internal constancy in professionals (professional point in figures) is measured by scoring the professional segmentation images pairwise against each other treating one then the other as ground truth. Scores are combined by finding the mean between professionals for each image then the mean between images. The same pairwise internal consistency was performed for novices. Other researchers have used the median between images but here we consider each image selected for training was equally important to parameter tuning.

## Boundary Metrics

Pixel Correspondence Figure J.4.2 For the creation of tactile diagrams, unlike in recent visual image segmentation (Figure D.8) one does not necessarily care about exact pixel correspondence, one-to-one match between boundary pixels (Figure J.4.2). The human tactile sense will not notice the few extra or few missing edge pixels from slight path differences. However, without accounting for pixel correspondence some algorithms may seem to perform better than other studies have shown because extra noise pixels near ground truth boundaries that were not excluded would boost the correct pixel count.

Precision Recall The common way of representing segmentation accuracy is with precision-recall (PR) plots. Precision and recall come from the confusion matrix formed by comparing predicted and actual boundaries. PR plots are parametric in nature where all the parameter sets represent the independent variable and the axes, precision and recall, are the dependent variables. When assessing the results of computer segmentation, the absolute truth for the boundaries is unknown, they are compared against human made boundaries. Traditionally the best performance is at 1-1 representing perfectly correct segmentation, however, comparing to multiple ground truths this becomes unattain-
able. PR is used over traditional measures of sensitivity and specificity because, in an image, the number of true negatives, labeled non-boundaries, from the blank non-boundary background would far overshadow all other data [51].

After comparing each ground truth segmentation against each other the mean score, is represented as a single professional point representing the target. Figure D. 8 shows an example of a cloud of human illustrated images reduced to a single human point and example PR plots from various segmentation algorithms. The Berkeley human point was projected to the new F-score in our results by viewing the F-score as a topology as seen in Figure D.9. This allows us to estimate what their F-score might have been without pixel correspondence.

F-Score The F-score is the harmonic mean of precision and recall. This non-linear form is chosen to express the increased difficulty of an incremental jump in score when scores are higher. For example, a student would find it more difficult on an exam to achieve a jump from 80-90\% compared to a jump from 20-30\%. The adjusted F-score accounts for random chance and scales the PR plane so that 0 is random while 1 is perfect. This involves knowing the anticipated random chance F-score. For assessments with two choices, such as ours where a pixel can either be a boundary or is not, one might assume $50 \%$ is random. However, since there is a much larger chance the pixel is background (i.e. non-boundary) the random chance score is actually close to zero. Most researchers, as well as us, report only F-score for boundaries. Unlike the other metrics Fscore cannot be combined between images and professionals with an average because of its non-linear nature. To handle this the mean PR scores are used to re-calculate the mean F-score.

Distance to Pro One concern with using the RP values as a method of assessment is that when multiple ground truths exist the 1-1 PR score is not longer attainable and there is often unequal balance between PR, unless the ground truths match each other
perfectly. This gave way to the potential for a new metric that does not look solely at the placement of the score on a PR plot but also the distance to the internal consistency between ground truths (human point), which we call Distance to Professional (D2P). This metric potentially captures the balance in the human point to be higher in one of either precision or recall as opposed to just looking towards unattainable perfect PR together. This means that if the professionals prefer precision with less noise and so do the predicted boundaries then they will better match together than general PR scores.

Loomis Distance Based on an understanding of the differences between tactile and visual perception, a new metric was used here, which we refer to as the Loomis distance (LD). The motivation for this metric was to account for an important difference between feeling and seeing images: namely that tactile spatial resolution is significantly poorer than visual spatial resolution. On the fingertip, touch has spatially often been described as a blurred version of vision. Assessment of this new metric consists of blurring the boundary image to represent the equivalent tactile sensation in a visual form (Eq. (J.4.4)) then calculating the difference between the images (Eq. (J.4.5)) [50]. This also allows for small differences in line placement to be ignored since they would not impact interpretation of the diagrams.

## Region Metric

Probabilistic Rand Index The next most common modern metric is probabilistic rand index (PRI), founded in information theory [52]. Multiple versions of the PRI exist in the literature but the one used here is simply the mean rand index from evaluating the predicted region/segment labels compared to the actual labels [51]. PRI is chosen over most label comparison metrics because those based in information theory do not require a match between labels (I.E., that the same label is used for corresponding regions/segments in each image). In segmentation, the areas may match but can have different
labels as labeling is arbitrary.

(a) 2009 [53]

(b) 2011 [51]

Figure D.8: Precision-Recall (PR) curves based on region boundaries comparing Berkeley's benchmarks. Previous work used the larger Berkeley image segmentation database and accounted for pixel correspondence resulting in lower curves [54]. Multiple human points represent the mean between illustrators for each image which are then combined by a median to get a single human point. Level set point located in bottom right figure as a purple dot labeled as ChanVese.


Figure D.9: F score viewed as a topology with flow lines indicated. This is used to help project human points from other research to an equivalent point laying along a new $F$ score.

## D.4.4.2 Results

Table D. 1 and Figure D. 10 show results from professional and novice internal consistency and comparing novices to professionals with professionals treated a ground truth. Figure D. 10 includes data from previous human segmentation from a group interested in visual segmentation. This data represents the median results for 30 illustrators [55]. Berkeley human point was projected to the new f-score in our results by viewing the F score as a topology as seen in Figure D.9. This allows us to estimate what their f-score might have been without pixel correspondence. The figure shows their original value and the new higher projected value. Results showed that novices had higher internal consistency among each other than professionals did when considering f-score but were nearly equal for the PRI and Loomis distance.

Figure D. 11 shows a typical example revealing key locations of an image where both professional and novice illustrators differed. Overall it was observed that, in many cases, the professionals took extra time to simplify the diagrams whereas the novices did not.

The professionals also often completed/filled-in incomplete/missing data such as parts of the aqueduct hiding behind trees or rocks.

To verify that there was no learning effect that might make the novices data internally inconsistent, their resulting scores over time were analyzed to determine if there was improvement over time. Figure D. 12 shows that there was not significant trend in the data indicative of a learning curve; this data revealed only a $0.1 \%$ slop in improvement which is overshadowed by the general variability.

Table D.1: Experiment Results: "Professional" and "Novice" represent internal consistency whereas N2P (Novice to Professional) is how well the novices matched the professional segmentation. Time for all methods represents how long the method took to complete (in minutes). The professionals self-reported $>1$ hour more time than shown here for their planning phase before beginning the segmentation, as well as spending additional time for applying texture to the graphics. Measures comparing professionals are based on a mean consistency because all illustrators are expected to have the same level of expertise, whereas the measures comparing novices based on the median to account for potential extremes. Standard deviation is represented with $\pm$.

| Metric | Perfect | Professional | Novice | N2P |
| :--- | :--- | :--- | :--- | :--- |
| F | 1.0 | $0.85 \pm 0.09$ | $0.92 \pm 0.08$ | 0.93 |
| D2P | 0.0 |  |  | 0.01 |
| PRI | 1.0 | $0.93 \pm 0.06$ | $0.92 \pm 0.06$ | 0.93 |
| Loomis | 0.0 | $45.4 \pm 22$ | $47.3 \pm 17$ | 45.5 |
| Time | $<1$ Min | 30 Min ${ }^{*}$ | 15 Min | 15 Min |



Figure D.10: PR space showing internal consistency for professionals and novices, and scores from novice treating professionals as ground truth. This plot also shows previously found human segmentation consistency projected forward to match f-score of professional segmentation consistency along side their original [55].


Figure D.11: Comparison of 3 novice segmentation to professional, highlighting key areas of difference.


Figure D.12: Comparison of internal consistency for professionals and novices and scores from novice treating professionals as ground truth throughout the image set as a function of image number. Most participants completed the images in order and there is no significant trend in improvement over time. Examples of the most inconsistent images are placed next to their corresponding drops in f -score.

## D.4.4.3 Discussion

The analysis showed that the professionals were consistent with each other which allows us to use their diagrams for algorithm optimization/training. Figure D. 10 shows initially startling results with the novices generating more consistent results with each other and better scores when scored against the professionals. This actually makes
sense when considering that the professionals spend more time simplifying the images to remove clutter and anything that could hinder the end user from interpreting the graphic. Since each professional removes or adds a considerable amount data (boundary lines) in their own way based on past experience their images differ more than the novices. The novices also tend to have more technical experience likely resulting in more consistent lines.

The novices scoring higher when using the professionals as ground truth could have indicated that the novice generated graphics were either nearly half way between the professionals or novices matched one or the other of the professionals more closely for different images. However, by examining the data the latter seemed the most likely to have occurred. For example, for the box turtle, one professional wanted to focus on the shape and one wanted to accent the markings to help convey the more specific species. In contrast, most of the novices only focused on the shape. This led to further discussion with the professionals to discover the degree of influence the context in which an image is presented in has over the diagram's creation. If the students have never seen a turtle and are first learning about, them then the overall shape is the most important. If they have learned about a turtle and then need to learn about the different kinds and their distinguishing marks then more detail is required. This poses an interesting dilemma for segmentation and visual to tactile diagram conversion: How should an unsupervised algorithm should balance detail with overall concept? For this thesis we will focus on the student's initial contact with a new image where overall shape will be most important. In subsequent work the use of multiple diagrams, some having more details, will be considered to convey more information about an image.

In comparing to previously collected data, our human professional internal constancy point seems considerably higher than for human segmentation data published by a research groups at Berkeley's. They seem to care more about precision at the expense of recall. However, this is not inconsistent as their are several differences between the two
studies. The most likely reason for this is difference is the absence of pixel correspondence in scoring for our results (as compared to previous human segmentation data). The secondary reason is the motivation/direction given to the raters. Our professionals were focused on conveying information to people who are blind or visually impaired, calling upon their similar training to influence how they segmented the images. Two additional reasons are the fact that out study only used two professional raters as compared to 30 raters with varied experience and the images came from a more focused genre of pictures and diagrams.

## D. 5 Algorithm Optimization

This experiment was designed to explore the parameter space for each of the three chosen segmentation algorithms (k-mean, level set, and gpb-owt-ucm) to determine the best performing parameter set for each. Further optimization was needed to that done in the literature for visual segmentation due to: 1) the significant modifications made to k mean and level set, 2) the difference of the previous images selected to ones commonly needed by individuals who are BVI, and 3) the outcomes needed to be focused on what is best for making tactile graphics, including the differences between tactile and visual perception.

The first step involved traditional parameter optimization in regards to six assessment metrics, including one that was specific to tactile diagrams (namely the Loomis distance). The next step was to evaluate the internal consistency of an algorithm with each of the resultant parameter sets (i.e. that repeated use of the algorithm with a given parameter set produced the same results). Internal consistency is an important component to consider for tactile diagrams because, with touch, users can primarily only explore the diagram serially. This is a time and memory intensive process. It is greatly aided by users forming hypothesis from local information to drive overall exploration. This is particularly true
if, based on local information, the user believes they have explored the diagram before. This will only be true if the real-time use of the algorithm produces a consistent output for a given diagram. A third step was used, based on a human comparison of similarity between the professionally created diagrams and the automated algorithms with the resultant parameter sets from the previous steps, as it was felt that the assessment metrics were not entirely effective in capturing what the key differences were for tactile graphics. The human comparison looked at how well the computer generated algorithms matched professional diagrams by considering how well line/boundary locations matched and if all the major components necessary to understanding the images were present.


Figure D.13: Segmentation optimization flow

## D.5.1 Parameter Tuning

Parameter tuning, based on assessment metrics that could be automatically calculated, was used to extensively explore the parameter space for the three different algorithms. This enabled steps 2 and 3 of the parameter optimization to be more tractable. For each of the different segmentation algorithms, a different group of parameters needed to be explored Table I.3.1. Segmentation performance, in this section, was determined through a combination of boundary and region assessment metrics, including one specific to tactile perception.

## D.5.1.1 Method

Image Selection Images were the same 37 training images as those used during rater reliability testing.

## Algorithms

K-means K-mean segmentation is restricted to feature sets that exist on a continuum leaving fewer feature options than for other algorithms, specifically color \& texture. The color sets explored were gray scale, RGB, Lab, and Luv. The texture extraction methods considered were local entropy or no method. Local energy was used over other texture extractions because it represents how much texture is in an area where as the others match more closely to a gradient of textures. While running, the algorithm matches each pixel to the closest cluster center in multidimensional feature space. The different calculations for determining the closest cluster were: euclidean distance, standardized euclidean distance (normalized by the cluster's standard deviation), dihedral angle, mahalanobis distance, and combinations of euclidean, standardized euclidean, and dihedral angle. This algorithm is particularity dependent on initialization state, so three initialization states were examined: uniform distribution, random sampling, and sub-sampling (run on $10 \%$ of the data to create initialization). Since k-mean requires knowing the number of clusters $(k)$ at run time a range of values from 2 to 8 was selected. A minimum value of 2 was selected because any less would no longer be segmentation and a maximum of 8 was selected as double the typical segments identified by professional diagram makers. This segmentation method was written in Matlab based on a built-in general k-means clustering algorithm and expanded for image segmentation and to handle additional distance metrics.

Level Set Level set has a great advantage over the other algorithms in the way it can include almost any type of feature set. Each feature can influence the 3D shape,
where features that represent a boundary push the shape towards zero and features that mark pixels that belong together push the shape away from zero together in an area. The specific features it can use to find segments are: spatial distance, color \& texture, gradient (energy), and contour (boundary) stiffness based on previous work [36]. These additional features are included here because unlike k-mean that is restricted to features represented on a continuum, level set can accept more abstract feature concepts. Like k-mean, level set attempts to determine pixel groupings based on likelihood they belong together (color space distance). All of the same color space distance metrics used to test k-mean were used with level set to help determine the likelihood a pixel belonged to a group. A weighting of each component and change in component weighting over time parameters were added to help fine tune how much preference was put on each of the fundamental components. Additional features explored were image space distance scaling, split/merge/cleaning parameters, and gradient parameters. Preliminary testing throughout the design process found parameter value ranges that produced usable images. These values were used with reasonable added higher/lower values included for final testing to capture the full range and account for interactions that may occur even at the extremes. Full parameter value ranges tested are listed in Appendix H.1.1.1.

Level set is also susceptible to inconsistencies due to initialization. The initialization states tested were: the solution to the optimized k mean algorithm, uniform distribution of circles, and a histogram based background detection with uniform circles. Specifically, parameters were adjusted based on how the uniformly distributed circles initial condition would perform given the circle's radius, padding between adjacent circles, and the shift between each phase's circles. Level set also requires knowing the number of phases ( $n$ ) at run time, so a range of 1 to 3 phases were examined $\left(k=2^{n}\right)$ to match the number of segments testing in k-means.

This segmentation method was written in Matlab based on the algorithm provided in [36] and expanded significantly to handle other color space distance metrics, other
color spaces, component weights, change in weights over time, gradient kernel sizes, gradient metrics, image distance scaling, and initialization states, as well as, a stopping function to determine when the solution has reached convergence (Section D.3.2). The weighted component and change in component weighting over time parameters were added to help fine tune how much preference was put on each of the fundamental components. Parameters dealing with how the algorithm was initialized were added to determine how well it would converge given different initialization and what is the best initialization configuration.
gPb-OWT-UCM The dominating contour detection algorithm uses global probability of a boundary with an oriented watershed transform (gpb-owt), created by Malik et al. At Berkeley [31]. This algorithm incorporates local gradients on each color dimension (energy), texture wavelets, and spectral analysis (eigenvalue analysis) to form a probability of a boundary. It is capable of accepting features in a form that predicts boundaries but not those that predict areas that belong together as a region/segment. This weakens its appropriateness for tactile diagrams, as it is the general region/segment shape that matters rather than precise boundaries. In addition, the region boundary information produced needs to be grouped into segments for future tactile texture application. Fortunately, the boundaries naturally always formed closed regions. Segments were estimated from regions based on similar color and texture within regions. Each region was assigned the mean color of the pixels contained within then the images were converted to indexed images allowing for a max of 6 indices/segments.

The gPb-owt-ucm algorithm has already been studied extensively and optimized by Malik and his colleagues with a large database of images, albeit for visual segmentation. It has become a standard to compare against as is. Therefore, rather than re-exploring the parameter space, we chose to use their optimized parameters with only the ability to set one high level parameter that represents how many boundaries and how complicated the output image is visually. Their optimized parameter set used: lab color space, textons
(a texture classifier), and gradients (multi-scale gradients on each color dimension and spectral analysis). We looked at varying the ucm high level parameter, $k$, between 0 and 1 to test the full range. This algorithm unlike the others does not require an initial number of segments. This algorithm does not iteratively reduce to convergence and does not require initial conditions. It was already optimized and it made sense to use as is rather than recreate and re-explore what was already explored.

Parameter sets explored Full list of each parameter values and level set break out test groups can be seen in Appendix H.1.1.1 with condensed list shown in Table I.3.1

Table D.2: Experimental constraints

| Data Set | 37 Common Images |  |  |
| :---: | :---: | :---: | :---: |
| Algorithms | K-Mean | Level Set | gPb-owt-ucm |
| Parameters | Color Space <br> Texture <br> Color Distance Metric Init Method Num of Clusters | Color Space | UCM Threshold |
|  |  | Texture | Preset Parameters: <br> Color Space <br> Color Scales <br> Number of Scales <br> Color Wt <br> Num of Spectra <br> Spectral Wt <br> Smooth Filter Coefs <br> Texture Classifier |
|  |  | Color Distance Metrics |  |
|  |  | Num of Phases |  |
|  |  | Component Wt |  |
|  |  | $\Delta$ Component Wt / t |  |
|  |  | Init Method |  |
|  |  | Init Circle Size |  |
|  |  | Init Circle Padding |  |
|  |  | Init Circle Shift |  |
|  |  | Merging Params |  |
|  |  | Splitting Params |  |
|  |  | Cleaning Params |  |
|  |  | Gradient Kernel |  |
|  |  | Params |  |
|  |  | Image Distance Scale |  |

Assessment Metrics Same region/boundary decomposition used in ground truth image preparation (Figure D.6) is used here to prepare segmented images for assessment. To begin with, PR plot curves were formed by altering a single parameter at a time, representing the effect of each change. Each point in Figure D.14a, indicates the result of a different parameter set for the corresponding algorithm. gPb-owt-ucm has the smallest number of points as it had only one parameter that was tuned. When multiple parame-
ters are altered together, the point cloud of all PR points from each parameter set can be condensed into a front (Figure D.14b). The front does not simply represent a single parameter change as with the points, instead it represents a potential best for that the current parameters being manipulated could achieve. Much like an ROC curve represents a tradeoff between either more sensitivity or more specificity this represents a tradeoff front between either more precision or more recall. In order to go from a parameter effect point cloud to a trade-off curve a front, the front was determined as the furthest points from the origin along the rays from the origin. Such tuning curves are shown in Figure D.8.

Each algorithm/parameter set combination was scored based on boundary and region metrics with the current most commonly used metrics being F-score and probabilistic rand index (PRI) [41]. The other assessment metrics that were used (As in Section D.4.4.1) were: distance to professionals (D2P), Loomis distance (Loomis), and time. In addition, due to the large number of parameters explored with level set, the risk of over fitting by selecting only the single best parameter set over all images was high. To reduce this risk and anticipate a large more diverse image pool a regression analysis was performed to estimate the best parameter set. Such a regression accounts for image outliers and parameter outliers to find their unique interactions and predict the best set overall. Through a general regression model and polynomial regression, statistical information can be gathered to find and predict the best parameter options. The effects of each parameter change on PRI scores were evaluated and the parameters that produced the highest PRI scores were selected. Raw regression plots are shown in Appendix H.1.1.2. In this scheme each parameter value and each image was used as an input and PRI score from each image and each parameter value as a target predicted output. The resulting metric is refereed to here as Reg-PRI meaning the PRI value for the parameter set combination selected by regression. In figures, measurements from regression are referred to as relative best. The regression method only focused on PRI because it had less noise and better predictive power.


Figure D.14: PR points: How well initial unaltered/unrefined algorithms and parameter sets match the professionals. Human point comes from professional diagram makers. Curves were not at the absolute front of the cloud at this early stage of evaluation they considered the mean of the furthest $10 \%$ of values from the origin.

Exploration Method Because of the large number of parameters that needed to be explored for level set, exploration of its parameter space was broken into smaller parts. Each part or group of parameters focused on specific subsets: base algorithm with color space metrics, component weights, initialization states, forced merge/split/stop, and noise reduction with image space and gradient methods. The parameters and values explored in each group are listed in Appendix H.1.1.1. Each successive level set test used the best parameters from the previous set based on D2P. Tests were run on multiple computers simultaneously, with each computer running up to 4 parallel concurrent tests in batch mode to complete testing in two months. To improve run time efficiency, each image size was reduced by $1 / 3$ in size, which is still normal printing resolution. Without splitting level set into smaller components testing every combination of parameters would have taken years to simulate on the same hardware. These smaller groups reasonably capture the most important configurations of the full test and are not expected to have any reduced value compared to testing all combinations. Each group was maintained through to the next phase of algorithm optimization because it was not yet known if D2P
was the best metric for basing each level of development on. K-means and gPb-owt-ucm were explored with the same hardware and with the same parallel processing, however, since they had far fewer parameters they did not need to be split into parts.

## D.5.1.2 Results

The comparison of the different algorithms in terms of time taken and the other scoring metrics is shown in Table I.3.2. Figure D. 15 shows the results of using PR curves to compare the three segmentation algorithms (gPb-owt-ucm, k -mean, level set). The figure also shows the points in the trade-off cloud that represent parameters that achieved the best score depending on the metric and algorithm used. The same score values can be

## seen in Table I.3.2.

Table D.3: Experiment Results: Best performance on each score metric from each algorithm, where the best parameter set could be different for each performance metric. "Professional" represents the internal consistency between the diagram made by the two professionals whereas each algorithm is how well it matches the professional segmentation. Time for all methods represents how long the method took to complete; it should be noted that gPb-owt-ucm was implemented mostly though C , which provides a speed boost over the other algorithms. The professionals self reported $>1$ hour more time than shown here for their planning phase before beginning the segmentation and additional time for applying texture to the graphics. Professional measures are based on a mean consistency because all illustrators are expected to have the same level of expertise.

| Metric | Perfect | Professional | K-Means | Level Set | gPb-owt |
| :--- | :--- | :--- | :--- | :--- | :--- |
| F | 1.0 | 0.85 | 0.80 | 0.81 | 0.82 |
| D2P | 0.0 |  | 0.007 | 0.003 | 0.007 |
| PRI | 1.0 | 0.93 | 0.80 | 0.82 | 0.81 |
| Loomis | 0.0 | 45.4 | 68.3 | 66.7 | 62.3 |
| Reg-PRI | 1.0 | 0.93 | 0.77 | 0.79 | 0.81 |
| Time | <1 Min | 30 Min * | 0.2 Sec | 20 Sec | 40 Sec |



Figure D.15: PR points: How close each algorithm and parameter set comes to matching the professionals and accounting for the inter-professional variability. Plot also includes markers for other points of interest. (PRI=Rand Index)


Figure D.16: PR curves: How close each algorithm and parameter set comes to matching the professionals and accounting for the inter-professional variability. Plot also includes markers for other points of interest. The curves more clearly show the precision/recall tradeoff of parameter change and the cross over between algorithms. (PRI=Rand Index)

We have included the results of the regression on PRI because it may provide a better parameter set for new images not presented. Sometimes the predicted best scores are slightly less than the scores found in PR space. Appendices H.1.1.2 to H.1.1.4 show the individual results of regression, the best parameter values based on each assessment metric, and example segmented images based on each assessment metric. Figure D. 17
shows how the parameter sets selected based on these additional inspection methods performed.


Figure D.17: PR points: How close each algorithm and parameter set comes to matching the professionals and accounting for the inter-professional variability. This includes the indication of where the parameter sets lie when not picking absolute global. (PRI=Rand Index, Relative=Reg-PRI)

## D.5.1.3 Discussion

Best Parameter Sets Each assessment metric identified a different best parameter set of values for each algorithm, and in the case of level set, each training group. Initial bests based on each metric are listed in Appendix H.1.1.3. With this reduced list of parameter sets we can not further refine and select the best parameters for tactile graphics in the following sections to continue optimization.

Feature Sets Perceptually based color was anticipated to perform better with all segmentation algorithms than other color spaces and they turned out to be the best. This makes sense because the human visual system processes images in this way on a regular basis so using these color spaces will help a computer see the same thing a person would. However, it should be noted that they were only marginally better than RGB (for k-means and level set, where they were compared to other color spaces) and there were a few select parameter combinations where gray scale performed better. Change in gra-
dient/energy parameters in level set seem to interact with other parameters making it unclear which parameters improve or hinder the algorithm's performance.

The addition of texture to the images feature space hindered k-means' and level set's ability to segment the images. Past research has shown that the inclusion of texture marginally helps, while other research focused on the removal of texture before segmentation [37, 43, 56, 57]. Their added benefit comes from attempting to identify a single simple shape around an object with a near uniform texture. Attempting to use a similar method with this research, interested in more detailed segmentation, results in significantly higher noise being introduced.

In terms of color distance metrics, Mahalanobis distance failed to produce the expected improvement. Mahalanobis should theoretically produce better results because of its unique ability to account for the interaction and non-independent nature of the color space dimensions. However, it ended up producing poorer results by over complicating the data and severely slowed the algorithm because of the system resources required to calculate the co-variance matrix on every iteration. Mahalanobis ends up locking on to much finer details multiplying the noise by capturing local extrema.

Assessment Metrics It was found that the boundary metrics, specifically those used to generate the PR points/curves, could be misleading. This is because these scores are based on how close the segmented boundaries match those generated by the professionals with an allowable margin of error and also allow pixels to be counted more than once. Some images may have noise adjacent to the actual boundary that get counted as extra correct pixels. The margin cannot be shrunk because it would count against boundary lines that are correct but might be too many pixels away from the professional lines. Other researchers have attempted to ensure each pixel is only accounted for once, but this leads to a unique set of problems. Luckily this noise is not too important because of the way the images are perceived tactually as seen with the Loomis filter.

It should be acknowledged that k-mean and level set are currently entirely imple-
mented in Matlab, which contains a known overhead: the time they take is expected to reduce even more if implemented in c/c++. gPb-owt-ucm has several components that were already implemented outside of Matlab, and so the time taken is not expected to change as much. For the given implementations, the fastest algorithm was k-means followed closely by level set and then gPb-owt, even though gpb-owt has components optimized outside of Matlab. The time taken by k-mean and level set would already be acceptable for generation and printing of tactile graphics and would also be acceptable for refreshable displays once they are implemented outside of Matlab to remove the overhead. The time taken for gPb-owt-ucm was not only much slower than the other two algorithms, it took a lot more computer resources (RAM and CPU cores) while requiring Linux. If we would assume the performance of a typical computer, as opposed to the one used in the experiment, the algorithm would take even more time. This would be acceptable for generation and printing of graphics but not for real-time refreshable displays.

This also shows how gPb-owt-ucm behaves differently from the other algorithms. It was also optimized by Malik and his colleagues using the Berkeley image database. Their optimization seems to reach for their human point, which has a higher percent of precision than recall. This also explains why the trade-off front curves for this algorithm crosses over each other as those algorithms reach for the professional point. This also indicates that potentially altering some of the lower level parameters in the algorithm may enhance how it performs with this our new set of images and intended use, particularly in that, because touch is less precise than vision, less of an emphasis on precision. Though such changes may refine the diagrams made by this algorithm they are not expected to significantly change the performance.

## D.5.2 Parameter Selection

Each image generated by the "best" parameters based on 5 assessment metrics has been visually inspected to determine the final parameters used to represent the algorithm group.

## D.5.2.1 Methods

As human cognition is still significantly better than any of the assessment metrics in determining how effective an algorithm/parameter set combination is in matching the results of the professionals, this step of optimization was performed by a human rater. For each image in the image set, visual comparison was made between the images generated by the algorithm/parameter set combination and the professionally created diagrams by a non-biased laboratory intern that was not aware of algorithm or the parameter set used. The rater was trained to follow three basic criteria: boundary lines should be in similar locations, boundary lines should split regions in a similar manor, and small amounts of noise near a boundary should be ignored. The intern rated an image for all parameter combinations (5) for each algorithm at a time until all algorithms (7) and images (37) were exhausted. For each image, the rater assigned 1 for the best match to the professionals and 5 as the worst. In the event of a tie because the images produced were exactly the same, they would both receive the same value and the subsequent number would be skipped. To put a heavier penalty on mismatch images if there was a tie for last place they would all receive 5. The best parameter set for an algorithm was determined by talking the median score across images for each parameter set of an algorithm and choosing the lowest median. In the event of a tied median the lowest mean was then selected. All five level set groups were combined together when calculating the mean and median to determine the best metric for overall level set. Each level set group was also considered alone to ensure simpson's paradox was not active when combining them together.

## D.5.2.2 Results

This step found that the parameter sets determined by the statistical regression and probabilistic rand index metrics were the best methods for determining accurate segmentation and boundary lines. Results of the visual inspection can be seen in Table D. 4 .

Table D.4: Shows which metrics performed the best when visually inspecting segmented boundary lines. Image were scored 1 (best) to 5 (Worst). All 5 level set groups are combined here. Values represent Mean :: Median.

| Metric | Overall | K-Means | Level Sets | gPb |
| :--- | :--- | :--- | :--- | :--- |
| Probabilistic Rand Index (PRI) | $1.93:: 1.57$ | $2.73:: 2.00$ | $1.79:: 1.40$ | $1.81:: 2.00$ |
| Regression (Reg-PRI, RG) | $2.08:: 1.71$ | $2.54:: 2.00$ | $2.15:: 1.80$ | $1.32:: 1.00$ |
| F-Score | $2.16:: 2.14$ | $3.00:: 3.00$ | $2.15:: 2.20$ | $1.35:: 1.00$ |
| Loomis Distance | $2.23:: 2.14$ | $3.59:: 4.00$ | $2.14:: 2.00$ | $1.32:: 1.00$ |
| Distance to Pro | $2.43:: 2.57$ | $3.57:: 4.00$ | $2.22:: 2.20$ | $2.38:: 3.00$ |

## D.5.2.3 Discussion

The potential best automatically produced images as determined from the parameter sets associated with the PRI and Reg-PRI metrics were rated the best and closely scored when looking at all algorithms. This required further investigation that revealed that the parameter set based on the PRI metric was significantly better for level set, while the parameter sets based on the Reg-PRI metrics were was better for k -means and gPb . Reg-PRI was likely better for k-means and gPb because it rejected outliers that negatively influenced PRI alone. There were likely fewer outliers with high leverage in level set regression analysis and over all a smaller score deviation per parameter set. The level set images based on the parameters from the PRI metric also had fewer missing lines that were deemed important to user understanding by our professional diagram makers compared to those based on parameters determined by the Reg-PRI metric.

Ideally it would have been desirable to find that only one metric produced the best parameter set for all algorithms. This would simplify the future exploration of new segmentation algorithms as they develop as we would have identified an appropriate and easily computable metric that correlated to the desired properties of the segmented im-
age for tactile diagrams. This was somewhat obtained as the regressed PRI metric does have some relation to the PRI metric which produced some consistency across algorithms. Ultimately, visual inspection will still ne needed to determine the most appropriate parameter set for making tactile diagrams, although this could be restricted to just the PRI metric and the Reg-PRI metric.

## D.5.3 Internal Consistency

An important evaluation metric came from determining if the methods resulted in the same segmentation every time. This was used to check if any algorithms did not converge to the same results every time which may require reevaluating the best parameters selected. In the event two parameter sets rank almost the same and one does not converge then the more stable alternative would be selected.

## D.5.3.1 Methods

Internal consistency was measured with F score, PRI, and time, for each algorithm's best absolute and relative parameters. Each parameter set was run 5 times sequentially on the same machine. Results were scored the same way as novices were scored based on pairwise internal consistency.

## D.5.3.2 Results

kMean was the most inconsistent with F-score ranging from 0.86-0.99 with a median of 0.99 , PRI ranging from $0.72-0.98$ with a median of 0.93 , and time difference under 1 second up to 11 seconds with a median under 1 second. LevelSet was fairly consistent with F-score ranging from 0.76-1 with a median of 1 , PRI ranging from 0.96-1 with a median of 1, and time difference under 1 second. Berkeley's gPb-owt-ucm was the most consistent with F-scores and PRI of 1 and time difference under 1 second.

## D.5.3.3 Discussion

K-means inconsistencies were due to parameters causing random initialization, when these parameters were not considered k-mean was almost as consistent as LevelSet. The inconsistencies in level set were due to k-mean initialization with a random component otherwise it performed as consistent as gPb-owt-ucm. Berkeley's gPb-owt-ucm was the most consistent because there is no random initialization. Any time difference under 1 second can likely be attributed to OS task scheduling. Based on these results unless a parameter set had outstanding results any that used a random initialization would not be used any further.

## D.5.4 Algorithm Selection

Previous studies in the lab have found that participants find difficulty in participating in experiments that are more than $3-4$ hours for 3 days in a row. We therefore had to reduce the number of algorithms tested to two plus professional made diagrams. Ultimately only two algorithms will move on to user testing and need to be selected. These were again selected based on visual comparison.

## D.5.4.1 Methods

As human cognition is still significantly better than any of the assessment metrics in determining how effective an algorithm/parameter set combination is in matching the results of the professionals, this step of optimization was performed by a human rater. For each image in the image set, visual comparison was made between the images generated by each algorithm and the professionally created diagrams by a non-biased laboratory intern that was not aware of algorithm or the parameter set used. The rater was trained to follow three basic criteria: boundary lines should be in similar locations, boundary lines should split regions in a similar manor, and small amounts of noise near
a boundary should be ignored. The intern rated each image (37) from each algorithm grouping (7), with each image produced from the parameters identified in Section D.5.2 until all images were exhausted. The rater assigned 1 as the best match to professionals and 7 as the worst. In the event of a tie because the images produced were the exact same they would both receive the same value and the subsequent number would be skipped. To put a heavier penalty on mismatch images if there was a tie for last place they would all receive 7. In the event multiple level set training groups were the best only one level set group would be selected for human testing and the next highest performing algorithm would be selected. The best parameter set for an algorithm was determined by talking the median score across images for each algorithm group and choosing the lowest median. In the event of a tied median the lowest mean was then selected.

## D.5.4.2 Results

Results of the visual comparison between the algorithms is shown in Table D.5. The level set algorithm group focused on determining initial conditions/state and gPb-owt-ucm were rated the highest.

Table D.5: Shows which algorithms performed the best when run based on their best parameter set found from PRI/Reg-PRI and visually inspecting segmented boundary lines. Images were scored 1 (best) to 7 (Worst). Confidence interval ( $95 \%$ ) in parentheses.

| Algorithm Set | Mean | Median |
| :--- | :--- | :--- |
| gPb-owt-ucm | $3.1(0.73)$ | 2.0 |
| Level set Init | $3.3(0.71)$ | 2.0 |
| Level set Noise | $3.2(0.68)$ | 3.0 |
| kMean Base | $3.2(0.64)$ | 3.0 |
| Level set MS | $4.4(0.82)$ | 5.0 |
| Level set Base | $5.1(0.52)$ | 5.0 |
| Level set Weight | $5.2(0.68)$ | 6.0 |

## D.5.4.3 Discussion

Contour detection with gPb-owt-ucm produced great results. Its limiting factors are: must be run in Linux and it does not produce segments it only find boundaries. Though
gPb-owt can also only be run in Linux and its time is acceptable for many uses, it is still far from real time especially when considering it has already been optimized to avoid Matlab overhead that the other algorithms are still currently bound by. It also out-performed the other algorithms based on several scoring metrics for segmentation accuracy and consistency, which means the slight increase in run time may be worth the improved performance.

Level set performs well enough so far but visually seems to need continued work to remove noise. Given that kMean and level set pixel probabilities (color distance) are similar, the increased noise in level set over kMean is likely due to level set's spatial components specifically from the gradient.
kMeans is too inconsistent and susceptible to noise. If the noise cannot be silenced with some basic techniques it will not be used in the remaining work.

Even with relatively good scores (within $15 \%$ of Pros) the images turn out under par which indicates that at this point the segmentation algorithm is less important and it is time to move to user testing to find the best segmentation then move onto simplification. Berkeley's gPb-owt-ucm algorithm seems to produce outstanding results but it requires Linux, abundant ram, and further processing to create segments.

After digital training, digital testing set was not performed. Observing that the metrics for scoring digital training are not compatible to user interpretation of results. Digital testing was skipped to make way for user testing.

## Selected Algorithms and Parameters

- gPb-owt-ucm
- Threshold - 0.2
- Level Set
- Color Space: Lab
- Number of Phases (n): 2 ( $2^{n}=k$ number of clusters)
- Distance Measure: Squared Euclidean
- Initialization Mask: Uniform Circles
* Circle Pixel Radius: 5
* Inter-Circle Pixel Padding: 17
* Pixel Shift Between Phases: 13
- Weighted Summation
* Combination Weighting: Gradient [1.5], Contour Stiffness [1.5], Pixel Color [1]
* Change in Weighting over iterations:Gradient [0.8], Contour Stiffness [0.8], Pixel Color [0.8]
- Gradient
* Spatial distance plateau scale: 1
* Gradient: Central difference
* Gradient color weight: [1, 1, 1]
* Gradient Gaussian Smoothing: kernel size 3x3, sigma 0.4
- Cleanup
* Clean: None
* Merge: None
* Split: None
* Determine Init number of segments: Static (set at start)


## D. 6 Usability Testing

## D.6.1 User Testing

User testing was performed to determine the effectiveness of the best of the modern segmentation techniques investigated, alone, in relaying information about computer generated graphics in tactile form. User performance in answering questions with the two best algorithms with optimized parameters were compared to performance with the
professionally generated diagrams. Performance was used for the most demanding user case: when a user is alone and is presented with a diagram to convey unfamiliar information that cannot be adequately expressed in words. This is the long-term goal of our work in providing independent access to diagrams for individuals who are BVI. This means that a teacher was not present to explain the diagram, as is typical in school for children who are BVI. It also means that labels, descriptions or specific questions that significantly cued the content of the images was also not included, as even if present, would not be informative for unfamiliar information.

## D.6.1.1 Method

Image Selection Images were selected for this study within two classes, photos and diagrams. Within each class, several groups were included such as: turtles, flowers, fruits, buildings, art, etc. Within each group were 3 variants (e.g. 3 different photos of turtles, 3 different diagrams of flowers, etc.). The variants in each group were selected by two TVIs to be of similar commonality and difficulty. A different variant within a given group was used for each algorithm being tested. To reduce learning effects in user testing while keeping the question difficulty similar. The image testing set can be viewed in Appendix L.2: this totaled 24 groups with 3 variants each, 12 groups from diagrams and 12 groups from photos. In addition, 3 training groups were used for gaining familiarity and fundamental awareness with each algorithm's presentation before testing of that algorithm commenced: 1 from diagrams and 2 from photos.

In the subsequent description of the study and results, algorithm 0 represents the professionally made diagrams. For parameter optimization, both professionals diagram were used for modeling; however, for user testing only a single version of each diagram was selected. The most representative diagram was selected as the "ground truth" by the professionals, together with the study staff. Algorithm 1 images were created using the gPb-owt-ucm algorithm with the optimized parameter set given in [41] and the high-level
parameter selected during our parameter optimization procedure. Algorithm 2 images were created from the developed level set segmentation algorithm described in this thesis based on the best parameters/setting found during parameter optimization. A different image set than that used in the parameter optimization was used in user testing to ensure model fit worked across common images not just the training set.

Participants Ten subjects participated. Participants were recruited through the Virginia Department for the Blind and Visually Impaired (DBVI) and the Columbia Lighthouse for the Blind (CLB). Participants were over the age of 18 and blind or visually impaired. Blindness ranged from congenitally blind to adventitiously blind, and from totally blind to low vision. Participants did not have multiple disabilities. Three subjects' data were unusable due to subjects not completing the required tasks. Subjects indicated that they did not use tactile diagrams in their lives and few used braille in their past with only one still using braille.

Environment Subjects were sat in a small quiet room with a desk between them and the researcher. Each participant was blindfolded to ensure no residual vision was used when interpreting the graphics. Subjects were encouraged to take a five minute break between each class of images and required to take a ten minute break between algorithms to reduce fatigue. Diagrams were presented in front of subjects on a metal plate and held in position with magnetic strips at the top and bottom of the page to prevent them from shifting position, ensuring each diagram was in the same location for each trial.

Trial Sequence Before any testing trials, subjects were allowed a familiarization period with one diagram made from the first algorithm they would encounter and a texture sample sheet. Before each testing algorithm block subjects were allowed two training images to practice with and discuss with the researcher. As with testing, the training images were from a group of variant triplets to prevent the same exact image from reappearing
in training and affecting learning.
Trials were blocked by algorithm and presented in a counterbalanced order across subjects. Within each of the three algorithm blocks, the image class (photo and diagram) was also presented in counterbalanced order across subjects. Within each group, the 12 images were presented in random order, and were selected in a counterbalanced manner from the variants within each image group. Image variants were used to reduce learning effects over blocks. For example, the food image group had 3 variants: apples, bananas, and ice cream. The variants were distributed randomly among the algorithms so that a subject was never presented with the same variant but still received images from all groups for each algorithm. Figure D. 18 shows a flow chart example of the testing setup. Users were give 2 example diagrams per algorithm where they were told what the image is.


Figure D.18: Shows how the experiment was designed with rectangles as variables, the line on the side of a block represents a between variable, the other blocks represent within variables.

Questions Subjects were asked to describe the shapes they were perceiving when feeling diagrams and after some time also asked to provide a category and name label for the image presented. This allowed use to assess the ability for participants to interpret unfamiliar information presented in diagrams, since this is the primary benefit for individuals who are BVI. More focused questions from previous studies of tactile diagram use (e.g. [9]) were not replicated here due to the potential for providing leading information and achieving ceiling effects. Real world use of the diagrams would also include braille labeling, descriptions, and, often, teacher involvement. Those were not included here to ensure the image processing algorithms were the only source of information being tested.

Subjects were encouraged to describe not only geometric shapes (circles, triangles, squares) but also curves, curve directions, smooth/jagged edges, points, plateaus, recesses, matching areas, etc. Subjects were asked to give each image a categorical label because visual studies found categories were important in building a schema for images to better interpret their content [58-62]. Professionals determined the name and suitable category labels for each image, which are shown in Appendix L.2. Subjects were asked to give the name label as it is the most precise, unambiguous description of the image. After each algorithm block, subjects were also asked to complete a systems usability survey (sus) to evaluate the usability of the diagrams in that set [63] to assess the usability of the algorithm method.

Prompts Before any testing trials, subjects were allowed some familiarization time with the type of tactile diagrams to be used. They received one diagram made from the first algorithm they would encounter to explore: during which the researcher discussed with the subject how to use tactile diagrams, what is depicted in the image and the questions that would be asked. Subjects were also encouraged to use multiple hands (the most effective method of using tactile diagrams) and were explained the importance of exploring the whole page. In addition, the researcher explained the line types and the textures used, the latter with the help of a sheet showing an example of each texture. They were told the textures were used to help distinguish parts from each other. Subjects were further told the images were of items commonly encountered by children at home and school, including in magazines, catalogs, science, geology, geography, and art textbooks.

Before each testing algorithm block, subjects were allowed two training images that used that algorithm to practice with and discuss with the researcher. The researcher allowed the subject to explore the image. They then reminded the subject of the questions to be answered and to explore the whole image. The subjects were then giving time to answer the questions when they were ready. Then the researcher told the subject what was depicted in the diagram.

Testing trials within each algorithm block were conducted so that the researcher only reminded the subject of the questions; they did not divulge the object in the image nor remind the subject to feel the whole image. If subjects requested a hint, the researcher would provide a short description indicating if the image was a single object, multiple objects, or a scene. The predetermined hints can be seen in Appendix L.2. Throughout testing subjects were not informs if the diagrams came from professionals or computer algorithms.

Scoring The correct answers for the category and name for each image was more easily determined than shape and were established before testing, with answers shown in Appendix L.2. Subjects received a 0 for incorrect answers and 1 for correct answers. Describing the shape of components of the image is more subjective, as parts do not necessarily have standard shapes. In order to ensure reliable ratings, two professionals (1 TVI, 1 researcher). Were asked to score the shape answers provided by the subjects as compared to the original visual image. The professionals were not aware of which algorithm was being scored, only the answers provided by the subject and the original image. They were asked to score each image based on how well the subject described the fundamental shapes required to understand the image. Subjects received a 0 if they did not describe any of the fundamental components required to understand an image and a 1 if they described them all, with increments of 0.25 for partial descriptions. The scores between raters were averaged.

Analysis SPSS was used to analyze the data via Generalized Estimating Equations (GEE). Algorithm, image class, algorithm order, and interactions between these main effects were included in the model. If they were found not to be significant, they were removed from the final model. The model was then used to determine the mean correct rate for each algorithm including the images prepared by the TVIs), mean correct rate 95\% confidence interval, relative risk between algorithms and professionals, and non-
inferiority between algorithms and professionals. The SUS was analyzed by converting the ratings to scores and calculating the mean with confidence interval between subjects. Scores below $<68$ are considered below average with a maximum score of 100 [63]. Both user understanding scores and sus scores were checked for rater reliability with Intraclass Correlation(icc).

## D.6.1.2 Results

Testing showed that 9 of the 24 image groups were outliers based on the wide disparity in the correct rate between variants for the professionally made diagrams. See image list Appendix L. 2 for grayed images representing excluded images. Table D. 6 and Figures D. 19 to D. 21 show results from the user testing with the different segmentation algorithms, excluding the outliers. These show that users were able to interpret the professionally made diagrams the best, with the level set algorithm second, and the gPb-owt-ucm algorithm last. Overall, however, both computer algorithms were determined to be non-inferior to the professionally made diagrams. Relative risk was almost the same between algorithms except for category naming which level set performed better for. Results from the SUS did not yield any significant difference between the automated algorithms and the professionals, but all scored below average. Table D. 7 shows which variables GEE found to be significant when modeling these systems.

Table D.6: Segmentation summary test results. This table shows how users performed when naming shapes perceived, categories interpreted, and object name between algorithms. It also shows how users rate each algorithm with system's usability survey. Relative risk indicates how many times more likely a user is to answer incorrectly for a particular algorithm compared to the professionally generated images (algorithm 0).

| Algorithms | Mean Correct (95\% CI) |  |  |  |  |  | ICC | Relative Risk |  | Non-Inferior |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  | 0 Pro |  | 1 gPb |  | 2 LS |  |  | 1 | 2 | 1 | 2 |
| Shape | 41\% | (11\%) | 23\% | (8\%) | 32\% | (10\%) | 0.18 | 1.2 | 1.2 | True | True |
| Category | 37\% | (12\%) | 17\% | (6\%) | 25\% | (10\%) | 0.32 | 1.7 | 1.1 | False | True |
| Name | 25\% | (13\%) | 13\% | (5\%) | 13\% | (06\%) | 0.20 | 1.6 | 1.6 | False | True |
| Usability | 42 | (22) | 35 | (20) | 36 | (22) | 0.01 | - | - | True | True |

Shape Mean Score


Figure D.19: Shows how users performed when listing shapes felt. Error bars indicate $95 \%$ confidence interval.


Figure D.20: Shows how users performed when listing category felt. Error bars indicate $95 \%$ confidence interval.

## Name Mean Score



Figure D.21: Shows how users performed when listing objects felt. Error bars indicate 95\% confidence interval.

Table D.7: Segmentation modeling variables found significant from GEE.

| Variable | Name | Category | Shape |
| :--- | :--- | :--- | :--- |
| Algorithm | X | X | X |
| Image Class | X | X | X |
| Algorithm Order |  |  | X |
| Algorithm * Image Class <br> Algorithm * Algorithm Order |  |  | X |

User Comments Users commented that, for tactile diagrams created by the gPb-owtucm algorithm, too many sections were left detached from the main body and the main body itself did not have enough details, which made the diagrams difficult to interpret. They also did not like that similar related components (such as flower petals) had dissimilar textures. Users were generally happier with the images created by the level set algorithm but suggested, to improve the diagrams created, the resultant images would need fewer blobs, less noise, less clutter from irreverent lines, thinner lines, and more detail related lines. Figure D. 22 shows example images that contain the components that subjects commented on. In general even for professional diagrams the users said the images were difficult to interpret. Users commented that textures were too strong and distracted from the boundary lines.


Figure D.22: Example images subjects commented on

Professional Comments Before user testing, when the professionals were allowed to see the tactile diagrams, they said they liked the produced images from all algorithms and only commented that images produced by the level set algorithm were noisy. After user testing and reviewing the results, the professionals provided further details about the algorithms. They explained that the gPb-owt-ucm algorithm had a problem in that it generated images lacking in consistency and which had too many small regions. They generally liked the images produced by the level set algorithm better, but thought it could benefit by including thinner lines separating the regions and using less salient textures. The professionals pointed out for both algorithms that in some instances there was too
much detail and in others key details were missing.

## D.6.1.3 Discussion

Subject performance was closest to the professionally made diagrams with the diagrams made by the level set algorithm, rather than the gpb-owt-ucm. ICC revealed that category was the most consistent among users. User performance and user comments indicated that more work is needed to further improve the computer generated tactile graphics. Improvements can now be made to level set to approach the user performance achieved when using the professional diagrams. When algorithm order was significant was significant to the model the middle test performed the lowest with the last test the highest. This might indicate fatigue where longer forced breaks are needed. In other cases when it was not significant later blocks were scored higher.

Limitations There were some limitations to the study. Perhaps the most significant limitation is that difficulty of interpreting the variants of an image was significantly different for 9 of the 24 images. Although it was random as to which variant was easier/more difficult than the others, to avoid this problem we only used images for which the scores for the variants were reasonably consistent with each other (and give numbers for what was reasonably consistent). For the question involving naming the object(s) in the diagram, it was noted that the distributions of the responses for all algorithms were considerably positively skewed. This suggests that flooring effects may have occurred, which may have hid some of the differences between the algorithms. This is supported by the fact that user performance for both the category question and shape question was higher for the level set method than the gPb-owt-ucm method. In addition, the distribution of the responses for tactile diagrams using the gPb -owt-ucm method for both naming the category and describing the shape was also significantly positive in skew, in contrast to the other response distributions which were more normal. Again, this probably created an overestimation in the performance of users using diagrams generated with the gPb-
owt-ucm method.

## Chapter E: Simplification

## E. 1 Introduction

Segmentation alone is not sufficient to produce tactile graphics similar to those created by experienced TVIs. TVIs first segment images mentally; however, they then decide which segments to remove from the picture and which to outline based on the importance and relevance of those segments to the content of the whole image. They often leave off many smaller regions that do not help interpret the image but will increase the cognitive burden for users when exploring the images: tactile perception of an image, even for basic pictures, already has a high mental demand which needs to be minimized as much as possible. Once the TVIs have outlined the relevant segments and regions, they add those ancillary lines that further improve understanding and add critical details: again, avoiding adding unhelpful information that will only add to the users cognitive burden. Finally, texture is added to only those regions for which the added texture will aid in comprehension of the diagram. Figure E. 1 shows example steps taken by the professionals.


Figure E.1: Example simplification images from a professional.

Supervised machine learning is a great statistical technique for modeling systems,
such as TVIs mental procedure for constructing tactile diagrams, where intrinsic information in a stimulus is used to predict the human response. Some advanced regression algorithms can handle hundreds of local and global information cues. This is of benefit for our purposes as how TVIs develop diagrams is a complex process that takes years for a TVI to become an expert. However, there have been limited studies of the decisionmaking process TVIs use to create these diagrams, and those studies have focused on simple mathematical graphs and charts. With the lack of any knowledge to narrow the exploration of the feature space that could be used in modeling photographs and diagrams, there are a significant number of cues that could reasonably be thought to contribute to the tactile diagram construction process.

## E. 2 Design Decisions

## E.2.1 Modeling

Modeling consisted of observing TVIs work flow and designing a computational work flow to match. Each phase in their decision process is broken out into its own statistical model, with the models then being merged together in a final step similar to TVIs. The feature extracted cues in the models come from varying amounts of processing of the original images Appendix L.1. The segment/regions results of segmentation from level set with boundaries/gradients from level set and gPb , are used to provide some of the cues. Google provides some cloud based machine learning algorithms focused on image specifics, such as facial recognition, from which the outputs are also used as cues for the simplification algorithms. In addition, low level cues like the ones used unconsciously by professionals are include in the model to improve accuracy. Many cues are explored in the literature review (Appendix J.1) but only the best are used here and listed in Appendix H.2.1.1.

## E.2.2 Components

Core modeling steps include finding edges that could represent either boundaries or ancillary lies, determining which segments/regions need to be removed, determining background areas that should not receive any lines, and determining which areas should be textures. The basic steps and how they are combined are showed in Figure E.2. Each model represented a pseudo-probability such as likelihood of an edge being present, likelihood of regions being accurate, and of texture being applied. Thinning reduces the width of predicted edges, cleaning removes superfluous regions, accenting combines thinned edges with region boundaries, and texture processing prepare texture to application to segments.

Some components of the process were not implemented computationally because they were expected to introduce more problems than they solved when based on stimulus cues only. The segmentation rejection model was not implemented because the segments as a whole do not need to be removed and were generally correct. However, the smaller superfluous regions within segments were the source of noise and need to be cleaned. Cleaned regions might intuitively improve edge anticipation accuracy and texture application. However, feeding cleaned region boundaries back into the edge model was expected to magnify confounds and noise while also overly suppressing ancillary lines. The edge model alone already accounts for raw regions and begins the cleaning process by internally marking low likelihood region boundaries as lower likelihood edges. A model to deem large areas as background that should not contain any lines was not implemented because often regions within the objects of an image had sparse lines similar to backgrounds and the model would suppress their desired details.


Figure E.2: Simplification model integration. Green components were considered but not used to prevent amplified noise and reduction of critical details. Double click to see larger diagram with example images for each step.

Edge Anticipation Edge anticipation is expected to: 1) model the way a professional diagram maker's brain decomposes changes across the images as edges (changes in continuous area) and ridges (lines between areas); and 2) then decides which of the edges and ridges will be included in their tactile graphics. This model must be tuned based on the actions of TVIs to include the correct amount of details so as not to overwhelm the user but contain enough details so that they can understand the graphic. It is particularly critical in ensuring the inclusion of ancillary lines that form critical details, and which are frequently removed by straight image segmentation methods.

This also provides an initial cross check against the region boundaries found from segmentation that are included as cues. Some region boundaries will be confirmed while others may begin showing cause for removal with lower edge likelihood. Conversely,
some region boundaries could also be used to confirm the presence of an edge. Given the sparse/bias nature of the diagrams with only a small potion of the image being desired edges, segmentation region boundaries are needed to help boost the model as an interaction term based on the features that comprise level set. Though various edge/ridge cues will be fed into the model it must decide which ones should be included in the tactile diagram based on how professionals make the same decision. Figure E. 1 shows an example of ridges that were missed from segmentation to form stitching.

Thinning Regardless of the type of methods used to develop an edge model, and whether an individual or combined methods is used, the edges/ridges on the resulting model diagram frequently need to be thinned. This is because all of these methods produce either spurious, wide, or jagged lines that need to be refined for user interaction and for further model construction. This effect was expected to be worse using machine learning models to generate the edges based off of TVI diagrams because they do not perfectly match their lines to image edges. Therefore a step is included to thin the edges before combining the results of the edge model with the other models developed in parallel.

Region Rejection Region rejection is expected to: 1) model the way a professional diagram maker's brain decomposes images into region components; and 2) then decides which of the regions will be included in their tactile graphics. Not all regions created from segmentation should be kept. Many smaller regions clustered together represent noise. Figure E. 11 shows an image that received many small interwoven regions from segmentation that needs to be cleaned. This model will help determine which regions are most likely in need of removal. Region addition is not explored because the segmentation testing already showed there was a need for rejection but not addition, that the needed regions are present but sometimes there are just too many.


Figure E.3: Example image needing simplification to remove small regions.

Texturing As with Figure E.1, the professionals added textures to help areas stand out from one another, when it is needed to improve the understanding of an image. For the previous user testing of the tactile diagrams produced by segmentation techniques alone, texture was simply applied to the largest few segments. However, many users, as well as comments by tvis who were shown the diagrams, thought there was too much texture, which distracted from the diagram. For understandable simplified images, only a small number of areas and only specific areas should get texture based on diagram: which areas are selected are based on a diagram maker's intuition. The diagram makers we talked to were not able to explain how they choose which areas should get texture; only that they took care to make sure not too many areas received texture. In some cases, professionals choose not to include texture at all in a particular image. Therefore, a model is needed to determine if there is a predictable pattern. Additionally the professionals suggested that users may have better understanding of the images if textures were not raised as high on the paper and if the textures were not applied fully to the edges of the areas they cover.

Machine Learning Method Machine learning methods in general were selected because they allow the computer program to consider many features and then attempt to reduce the number of coefficients, and thereby the number of features, needed to mimic human processing of the images into tactile diagrams. Traditional linear regression was
not used because the response variable (whether the pixel should be raised or not) is not a normal distribution, as the pixels, on average, are highly biased to be background, and input features may be codependent. It was also expected that errors were not normally distributed and variances non-homogeneous. These all violate the assumptions made in using linear regression. In contrast, neural networks can sift out codependency and do not require normal distributions. However, neural networks were not used because they are computationally expensive, are likely to over-fit data with small sample sizes (such as the limited number of tactile diagrams we could get TVIs to create), and highly biased data (such as with most pixels being background) can still negatively affect modeling. Generalized Linear Models (GLM) do not have the same assumptions and constraints as linear regression and neural networks making it a great candidate to use. All of the methods including GLM would work much better with data that is not as biased as the data here with only $6 \%$ of the data representing lines/edges/ridges. Therefore, a mechanism was sought to try and transform the data so that it not quite as biased.

Cue Selection For each process model, an initial large set of feature possibilities, explored in Appendix J. 1 with example results, was reduced to a select few features based on their individual performance. Though far more cues were examined, the best ones for a target range of performance over many image types were preferred. Many overlapping cues were maintained because they each showed merit under different common circumstances. Each one needed to be further examined for a wider range of potential input images within models attempting to predict diagrams as professionals would. Various levels of features were considered from local to regional to segmental to global. Local cues were expected to be the biggest influences on edges but other levels could still have an effect or even interact. For example an image that is mostly tinted red would have an overall global image color shifted towards red and could change how lines are predicted. Region and segment cues are expected to have the largest influence on region rejection but others such as many large local gradients throughout the image might indicate fewer
regions should be trusted.

## E. 3 Design Implementation

## E.3.1 Cue Extraction

Eq. (E.1) shows the base model formula in Wilkinson Notation used for each model. Appendix H.2.1.1 shows the complete list of all potential cues with the condensed list shown in Figure E.4. Abbreviations such as cues means each cue is included directly. Notation such as local_energy ${ }^{2}$ means each local energy cue is also included as a quadratic term and local_energy : region_energy indicates the interaction terms between each local energy cue and each region energy cue are included. Braces indicate each cue category is accounted for in the interaction terms. Appendix H.2.1.1 lists 225 cues which become 3447 terms after included the intercept, quadratics, and interactions. Using that many terms is not sustainable for most regression techniques and potential correlation between cues will weaken the resulting model predictability. Future applications of this project also require a user to be able to interact with images in real time which would not be possible if all 225 cues need to be extracted from every image. Therefore terms need to be reduced further to only the most significant.

$$
\begin{align*}
\text { Base }= & 1+\text { cues } \\
\text { Expanded }= & 1+\text { cues }+ \text { local_energy }{ }^{2}+\text { local_distance }^{2}+ \\
& \text { region_energy }{ }^{2}+\text { segment_energy }{ }^{2}+\text { interactions } \\
\text { interactions }= & \left\{\begin{array}{c}
\text { local_energy } \\
\text { local_texture } \\
\text { local_edge } \\
\text { local_distance }
\end{array}\right\}:\left\{\begin{array}{c}
\text { region_energy } \\
\text { region_position } \\
\text { segment_energy } \\
\text { segment_position } \\
\text { global_color }
\end{array}\right\}+  \tag{E.1}\\
& \left\{\begin{array}{c}
\text { region_energy } \\
\text { region_position }
\end{array}\right\}:\left\{\begin{array}{c}
\text { segment_energy } \\
\text { segment_position }
\end{array}\right\}
\end{align*}
$$




## E.3.2 Cue/Observation Preparation

A small experiment was designed to evaluate the effectivness of using different feature selection algorithms to find the most suitable for our data set. For feature selection, the Least Absolute Shrinkage and Selection Operator (LASSO), Partial lease squares (PLS), and Principal Component Analysis (PCA) were considered. Though other methods exist for feature selection they do not work well with sparse highly correlated cues. Pre-regression scaling techniques considered, to deal with variations in the ranges different cues, were normalization, standardization, and standardization with trimmed outliers. Post regression outlier rejection methods considered were large leverage and large cook's distance.

## E.3.2.1 Method

To test the modeling techniques before implementing all of the required models a cursory edge anticipation model as in Section E.3.3 was used. The same images and ground truths as use in segmentation training were used here.

Pre-Regression Preparation Many of the cues produced were on widely different scales. Scaling was needed so all cues were on a level playing field. This prevented finding large prediction coefficients simply because the cues was tiny. It also prevented loss of influential cues just because they were large and the resulting coefficient was small. The methods evaluated were: normalization, standardization and standardization with trimmed outliers. The overall flow of the methods tested is shown in Figure E.5. Normalization was accomplished by rescaling the cue range to either -1 to 1 or 0 to 1 , depending on the initial starting ranges. Logical cues were not altered. Standardization was done after normalization by shifting values so that the mean was at 0 and the standard deviation was at 1 . Due to the overwhelming 0 bias in many cues, values within 0.05 of 0 were not included when calculating the mean or standard deviation but still shifted/s-
caled with the rest of the data. Values further than 0.95 away from 0 were considered outliers and also not included when calculating the mean or standard deviation. Another version of standardization came from calculating the mean and standard deviation after an initial regression and outlier rejection process was done. Ground truth cues were only normalized with no outliers removed at this point.


Figure E.5: Simplification cue scaling alternatives

Assessment Metrics Often regression models are assessed with $R^{2}$ to determine how well the model fits ground truth. To improve this metric, the adjusted $R^{2}$ (Adj $R^{2}$ ) was used because it accounts for the use of different cues depending on which selection method was used and different number of observations depending on outliers. Adj $R^{2}$ values close to 1 are perfect correlation and values below one indicate worse performance. However, Adj $R^{2}$ alone is not a sufficient assessment of performance. Since pixels in the edge model were biased heavily towards background due to the nature of the data, outliers needed to be checked to ensure the model was not simply rejecting all edges. Before outliers were rejected, the ratio of background to edges was checked to ensure it matched closely with the ratio of edges in professional diagrams. The edge outlier ratio is presented as the percent of edges considered outliers for the given model. Although this may sound complex, removing a few outliers helps prevent over fitting while removing too many outliers makes the model unable to work for a wide variety of images. Finally, in order to determine if any critical terms were missing from final models the residuals and
fits were examined for patterns that would indicate unaccounted for variances.

Cue Selection To reduce the cues of interest to a manageable set different feature selection methods were examined: PCR, LASSO and PLS. Feature selection works by selecting the most important cues and does not consider interactions yet. After initial cue selection, cues were further reduced based on results of GLM cue significance and influence as in Section E.3.3. Influence was interpreted first as the absolute value of the largest weightings then inclusion of only $p$ values $<0.001$. The number of top cues was ranged from 10-40 such that with interaction/quadratic terms GLM converged to the best Adj $R^{2}$ value. With more than 40 cues GLM no longer converged and fewer than 10 cues produced poor Adj $R^{2}$ values.

PCR PCA as part of principal component regression (PCR) was examined with a general linearized model (GLM) as the backbone regression. This method involved finding the principle components, selecting only the most important/influential cues based on these components, then using the selected cues in a GLM regression model. PCA accounts for the variance within cues to find the variables that explain most of the model input variance [64]. The principal components themselves are not regressed because they do not include the interaction terms. Important cues from PCA were selected by using the top 120 largest absolute values of cue weightings from the top 4 principal components.

LASSO Lasso was also used to find the most important cues. Though lasso is a regression model on its own, it is considered biased due to the regularization method. Therefore, the cues found from lasso were then used with a GLM regression model to prevent bias in the weightings of the model terms [65]. In evaluating the effectiveness of LASSO, only normalized, not standardized, cue preparation was used in accordance with LASSO requirements [65]. LASSO also makes use of cross validation to ensure that
with different data subsets it selects the same cues. A three-fold cross validation was implemented. LASSO iteratively works by allowing more cues in subsequent iterations based on constraining functions (Eq. (J.3.2)). The parameters selected were from the model with the lowest mean square error (MSE).

PLS Partial least squares (PSL) was also considered because it takes PCA further by not only considering input variance but also output variance and acts like LASSO in iteratively including terms [64]. This is achieved by projecting the input and output data into a new space then only including the first few components much like the first few PCA components. Due to the precise combination of cues within each component, selection of cues to use outside of PLS in a GLM model, the same way they were selected for PCR, was not possible as it would not converge. This means all cues are necessary every time the model would be used to make predictions on new images which would greatly increasing run time in future applications. Even with this limitation, PLS was explored due to its achievements in other applications [64].

Post-Regression Outliers Additional outliers were found after initial regression. A combination of leverage, cook's distance, residuals, and residual patterns was examined. Leverage alone was not used since cook's distance improves upon it. Cook's distance represents the change in regression results based on removing a single observation (single pixel) due to its leverage over the entire model [66,67]. Observations with large cook's distance (above 4 times the mean) and observations with large residuals (above 50\%) were excluded from regression to prevent over fitting to those observations with high leverage. The overall process is shown in Figure E.6. If any image accounted for more than $10 \%$ of the excluded outlying observations then the entire image was excluded.


Figure E.6: Simplification outlier removal flow

## E.3.2.2 Results

None of the models showed meaningful patterns in residuals and fits to indicate missing cues. Normalized PCA scored marginally better than LASSO with standardized PCA scoring the highest. Table E. 1 shows experiment scores. Overall LASSO selected fewer cues than PCA. Standardized models showed higher outliers than normalized models.

Table E.1: Simplification cue preparations. There are normally more terms than cues because of the addition of quadratic/interaction terms. PLS uses all cues in a new feature space and only selects the top terms in the new space. Outliers is the percent of pixels marked as outliers compared to the total number of pixels. Edge Outliers is the percent of edge pixels marked as outliers compared to the total number of edge pixels.

| Method | Adj R | Outliers | Cues | Terms | Edge Outliers |
| :--- | ---: | ---: | ---: | ---: | ---: |
| Professionals to each other | 0.30 | - | - | - | - |
| PCA, Normalized | 0.84 | $7.48 \%$ | 31 | 105 | $9.5 \%$ |
| PCA, Standardized, Pre-regressed | 0.94 | $30.90 \%$ | 20 | 85 | $57.9 \%$ |
| PCA, Standardized, Trimmed | 0.97 | $28.21 \%$ | 28 | 102 | $56.4 \%$ |
| LASSO, Normalized | 0.83 | $8.89 \%$ | 14 | 37 | $0.2 \%$ |
| PLS, Normalized | 0.65 | - | 225 | 10 | - |

## E.3.2.3 Discussion

This evaluation was used to discover the best method for modeling data for this application. PLS scored low and requires all cues to be calculated to create the new feature space to implement the prediction model. PCA using standardized cue values, appears to show the best Adj $R^{2}$ score which on its own would indicate that it should be selected.

However, although the Adj $R^{2}$ takes into account the number of observations and cues, it does not take into account any bias in the data. This is particularly problematic for our data as a small portion of the image pixels are critical for interpreting the diagrams (e.g. pixels that are part of edges in particular). If they are removed from the model as outliers, then the model fit will not convey the desired properties in the tactile diagrams. Therefore, the percent of outliers must be considered. The standardized methods have a large percent of outliers and an extreme percent of edges marked as outliers. Having such a high percent of edges not included in the model would make the model even more biased towards background, not stand up well for a range of images, and even not handle edges well in general. This means that standardized models are not acceptable.

Better results are achieved by normalized PCA and normalized lasso when considering all assessment metrics. Between normalized PCA and LASSO the Adj R $^{2}$ scores and general percent outliers are essentially the same leaving the decision to the percent of outliers that are edges and the number of terms. LASSO shows a much smaller percent of edges considered outliers. It also has considerably fewer cues to produce the model. The latter is important as in the future end use of this work, we want the computer to provide the users tactile graphics in real-time interactions with minimum delay. Using fewer cues will allow for faster operations. As there were no patterns in the residuals, this suggests that the fewer cues were not achieved by missing critical cues.

Therefore, LASSO is the best feature/cue selection method for this work. It also should be noted that since PCA explains input variance not necessary output variance, it may pick variables simply because they have more chaos not because they are actually important to the model.

## E.3.3 Edge Anticipation

Many of the tactile diagrams generated by image segmentation algorithms lacked ancillary lines after segmentation that are present in professionally made diagrams. To add
ancillary lines to the diagrams, as well as keep desirable boundary lines, a machine learning model was created. This model attempted to predict the probability at each pixel that a professional diagram maker would mark that pixel as an edge/ridge/boundary based on image cues. Following the procedure refined in Section E.3.2, cues were first normalized, then lasso was used to reduce the set of input cues, followed by GLM. Some cues were considered for being represented in interacting terms or quadratic form based on the likelihood they would contribute to the model; however, not all were used to conserve system resources and avoid over fitting.

Due to the limited spatial specificity of touch, often the professionals were not concerned with exact placement of edges but favored close enough lines for speed. To accommodate slight shifts in the results that should not effect the model, a Loomis filter that widened [thickened?] edges by blurring was applied to the targets. This was used to represent the lower spatial resolution of human touch and to robustly merge multiple professional's work when averaged together. The blurring also helped to reduce the model bias for non-edges by changing the number of pixels from 6\% binary edges to $34 \%$ scalar edges (the blurred edges make a pseudo-probability of edges). The overall process is shown in Figure E.7. Appendix H.2.2 shows the results for the following steps of the model: lasso, leverage, cook's distance, residuals, fit, outliers, score, and significant cues/terms. Figure E. 8 shows example images produced by the overall edge anticipation model. In order to apply the model to testing images, the same normalization parameters from training were applied to normalize testing cues.


Figure E.7: Simplification edge anticipation model. Base comes from base cues and Expanded comes from Expanded cues.


Figure E.8: Example images from edge model. Shows original images followed by predicted edges.

## E.3.4 Thinning

As the edge anticipation model predicts the thick visual representation of thin tactile lines, the thin lines need to be recovered for use in tactile graphics. Various methods for thinning edges produced by the edge anticipation model were designed and tested. This included a variety of techniques in Appendix J.1, such as: smoothing operators, inverse filters, deconvolution, hysteresis threshold, thinning morphological operators, and non-maximum suppression. Figure E. 9 shows the final procedure. For the first step, the Savitzky-Golay filter was used as a smoothing filter to reduce noise prior to deconvoIution. It was chosen as it does not blur lines like Gaussian filters or many other point spread functions [67]. The next step was to deconvolve the image to take the edges predicted from matching the Loomis filtered professional edges (Eq. (J.4.4)) back to normal edges. Standard deconvolution was attempted; however, it was unable to handle modeling noise and introduced more noise in the process. In contrast, the RichardsonLucy deconvolution was able to nearly recover edges even with modeling noise, due to its iterative nature [67].

As the lines could benefit from further thinning, an additional step was considered
using different techniques. Morphological operators were considered for this step but all had drawbacks for our scalar images as they work better with binary values: the skeleton operator thins lines towards the middle not necessary the maximum values; erode removed too much information in lighter areas and not enough in darker areas; generic thinning removed ancillary lines; non-maximum suppression left spurs; and despurring only removed tinny spurs $[67,68]$. However, non-maximum suppression was selected as it did further thin the lines without removing critical information like the other operators. In order to remove all light irrelevant edges that would feel like printed artifacts, the resulting images were quantized so that pixels fell into 4 equal buckets and low 2 values were removed. To further remove light spurious components, a mild Gaussian blur was applied. Figure E. 10 shows example images resulting from the developed thinning procedure.


Figure E.9: Simplification edge thinning process.


Figure E.10: Example images from edge thinning. Shows deconvolution of Loomis filter on predicted edges followed by thinned edges.

## E.3.5 Region Rejection

Many regions produced by even the optimized segmentation algorithms add superfluous information to diagrams and needed to be removed to avoid users experiencing cognitive overload. To predict which regions should be removed a machine learning model was developed with the implementation shown in Figure E.11, following the procedure refined in Section E.3.2. The target model response was produced by finding the Jaccard index (Tanimoto coefficient, ) between regions from each diagram made for an image by each of the two professionals with the regions found by the optimized level algorithm. The two scores where then combined via a mean. This results in a score representing a pseudo-probability of the level set region matching the identified regions in professional images. The resulting prediction from the model gives a method for estimating the likelihood a region produced by the optimized level set algorithm should remain, or conversely should be removed based on cues from the image.


Figure E.11: Simplification Region model.


Figure E.12: Example images from edge thinning. Shows region boundaries followed by score showing how likely region exists in images made by professionals.

## E.3.6 Region Cleanup

The steps for removing regions that were rejected by the region rejection model are shown in Figure E.13. Regions with a predicted low (<0.05) Jaccard index were absorbed into the surrounding regions. This was performed by modifying an image impute method (meant to replace missing data with substitute values) to replace pixel region labels of those regions being removed. The algorithm was modified to use the mode of surrounding pixel labels not the traditional mean [67] as mean does not work for nominal labels.

After the relabeling, region labels were converted to boundaries in preparation for combining them with edges Figure E.13. Then the boundary images were smoothed with a Savitzky-Golay filter. As gray scale printing on microcapsule paper does not linearly match gray color depth with the height of printed line the next step was to correct the gray scale for printing: boundaries were further refined with an error function as seen in $A p-$ pendix J. 1 to non-linearly map current values with plateaued ceiling and floor (sigmoid). The cleanup process is shown in Figure E. 13.


Figure E.13: Simplification region rejection cleanup.


Figure E.14: Example images from edge thinning. Shows cleaned regions.

## E.3.7 Accenting

The idea behind this step is that the edge model and region model capture lines differently depending on the image content. This process was developed to take advantage of the fact that lines that have a high probability of existing in both models are more likely to be true lines than lines that are not. The algorithm steps are given in Figure E.15. From the previous step of region removal, the boundary images indicated probability as gray scale that was adjusted for printing on microcapusle paper. To combine the models, first the same error function as used for boundaries was used to match the resulting edge model gray scale image to micro-capsule paper. In both cases, the resulting values of the pixels were between 0 (as background) and 1 (as full height edge).

The pixel probabilities of the region boundaries were floored to 0.5 and further squared to take into account the unreliability of the calculation as the probability decreases into noise, resulting in a range of $0.25-1$. Resulting region boundaries were combined with edge values via the calculation of their joint probability (multiplication) producing accented lines that include ancillary lines. Figure E. 16 shows example images from accenting. This
means a region boundary not present in thinned edges is removed while lines in thinned edges but not regions boundaries are reduced in height and lines in both receive full height. This non-linear combination produced the best diagrams, balancing edge and region information for our variety of images.


Figure E.15: Simplification accenting process.


Figure E.16: Example images from edge accenting. Shows thinned edges followed by accented edges.

## E.3.8 Segment Texture Model

Similar statistical modeling as for the other components of the tactile diagram formation process was explored to determine when to fill in segments with texture that would best approximate the decision process of professional diagram makers. The texture likelihood model is shown in Figure E.17, following the procedure refined in Section E.3.2. The target predicting response was found by marking each segment each professional applied texture in an image with a probability of 1 and all other segments with 0 . Then the probabilities from both professionals were averaged, producing a likelihood of each pixel of an image receiving a texture. Next, all pixels of the likelihood map that corresponded to being within the same level set generated segment were averaged together to represent a likelihood that the segment would receive a texture.


Figure E.17: Simplification Texture model.


Figure E.18: Example images from edge accenting. Shows segment boundaries followed by score showing how likely segment has texture in images made by professionals.

## E.3.9 Segment Texture Application

Two different versions of applying texture were tried. The first simply applies texture to the accented image. The second, based on recommendations from users and TVIs, distances the texture from the edge lines and reduces their height compared to the lines. Figure E. 19 shows the two version of texture application that follow the texture likelihood prediction to produce the final simplified images. They both find the segments most likely to receive textures above 0.4 likelihood, up to three segments.

The first method simply masks the textures and applies them to the desired segments. The second method scales each texture to produce lower line heights when printed, based on the best height selected for each texture by professionals. Each texture is again masked to match the area defined by each segment that will receive a texture. The second method also applies an erode morphological operator to each texture mask to produce a buffer space between textures and edge/boundary lines to help users distinguish lines from textures. Then it applies the textures to the accented image. Figure E. 20 shows example resulting simplified images.


Figure E.19: Simplification texture application flow.


Figure E.20: Example images from simplification. Shows texture scores followed by simplified ${ }_{1}$ images with texture and simplified ${ }_{2}$ images with textur.

## E.3.10 discussion

It was observed that for some diagrams there were too many edges while for images there were not enough. The continuum of too few to too many edges seems to be normally distributed. This also seemed to be because of variations between diagrams in what should or should not be kept, depending on the meaning of the content and how
professionals interpreted the diagrams. However, this cannot be dealt with unless we can take into account the meaning of the content, which is beyond this thesis. This suggests that what was achieved here was the best that can be done by using only the stimulus itself since all reasonable cues were extracted and included. We would expect there to be a balance of diagrams with too many edges to those with too few edges.

## E. 4 Usability Testing

User testing assessed the ability of users who were blind or visually impaired to answer questions about the tactile diagrams with which they were presented. The first study recruited individuals from the general population of individuals who are BVI, most of whom had little or no experience with tactile graphics. Due to significant differences in performance that were observed between study participants who had notable experience with graphics and those who had none to little, a second study was performed. This study only used individuals who had notable experience with tactile graphics as they are the most likely candidates for the independent use of tactile graphics. This is where the rapid generation of tactile graphics without any intervention by a TVIs is most needed. Automatic generation of tactile graphics would also be beneficial even with TVIs available as it would save them significant work when generating the graphics. However, those diagrams are normally explored with the guidance of a TVI.

The use of four different methods of generating tactile diagrams was compared. The standard for comparison was the set of diagrams made by the TVIs. The second method was the best performing method that used segmentation only: this was the optimized level set method. The third and fourth methods both used the simplification process described earlier in this chapter. The difference created between the latter to methods was due to the significant comments on the previous user testing as to the texturizing of the diagram influencing the user's perception. One of the method's texturized segments
completely while the other adjusted the height of the textures and had a blank buffer between the texture of a segment and the segment's boarders Figure E.20.

## E.4.1 Image Selection

The same picture set was used as in the segmentation algorithms user study, where pictures were of objects and scenes found in schoolwork and everyday living. There were 24 images grouped in comparable triplets. Half the images were from photographs and half the diagrams were from diagrams. In the study, tactile diagrams generated by algorithm 0 (Pro) represented professionally made diagrams. Algorithm 2 images were created from level set (LS) segmentation alone, as used in the segmentation methods assessment. Algorithm 4 (Simp) and algorithm 5 (SimpT) images were created from level set segmentation plus a simplification model produced by modeling the design process of professionals. For algorithm 4, texturization used a blank buffer between the textured area and core boundary lines, with the textures lower in height. Algorithm 5 used textures that were full height, with no buffer blank space between textures and core boundary lines.

Different image sets were used for testing than were used for training to ensure model fit worked across common images not just the training set.

## E.4.2 User Testing 1

This test was designed to compare the performance of users who were BVI with three of the different tactile diagram methods described above: algorithm 1 (Pro), algorithm 2 (LS) and algorithm 4 (Simp). This study drew from the general population of individuals who were BVI.

## E.4.2.1 Method

Participants Different test subjects were used than those who were in the segmentation methods assessment study. Participants were required to meet the same requirements as for the segmentation study. Twelve subjects participated. Four subjects' data was unusable due to subjects not completing the required tasks. Participants were over the age of 18 and blind or visually impaired. Blindness ranged from congenitally blind to adventitiously blind, and from totally blind to low vision. Participants did not have multiple disabilities. Most subjects indicated that they did not use tactile diagrams nor braille in their lives, with only three having used tactile graphics in their past and still using braille in their lives.

Environment The same set-up was used as for user testing of the segmentation methods.

Trial Sequence Images were presented in nested blocked groups based on algorithm ( 0,2 or 4 ) and then image type (diagram, photo). Blocks were counterbalanced between subjects. Images within each block were randomly presented, where each image was selected from three variants of that image group (1, 2 or 3). Image groups were used to prevent learning over blocks by repeated exposure to the same images.

Prompts Training was similar to segmentation but, before each algorithm group, users were told about nuances of that algorithm to simulate foreknowledge users would have in real world use. Before the level set algorithm group, they were told the image might be noisy with small areas that feel velvety or small detached areas. Before the simplified algorithm group, they were informed there may be small spiny protrusions from the lines.

Procedure The experimental procedure was the same as in the user study of the segmentation algorithms but with different algorithms and additional questions. Users were
asked to focus on describing the shapes as the most important question, while still attempting to categorize and name images. A modified form of the NASA task load index (TLX) question set was asked after each image. This consisted of asking users to subjectively gauge their perceptual load (difficulty perceiving lines, textures, and shapes), cognitive load (difficulty combining the elements into something meaningful), frustration level, confidence in answers, time to completion on a scale of 1 to 5 .

After users performed the designed experiment, they were given an opportunity to explore the 3 methods side by side to rank and discuss what features they liked or did not like. Ten images were compared in turn, where the resulting tactile diagrams from each of the algorithms was presented in random order in front of the study participant. They first ranked the image in terms of ease of understanding without knowing what the image represented. They then ranked the images again, once they were told what was in the images. Ranking is from 1 as highest preference to 3 as lowest preference.

Scoring Scores for name, category, shape and usability were determined in the same manner as in the user study of the segmentation algorithms.

Analysis The analysis performed was similar to what was done with segmentation user study using SPSS GEE modeling. For analyzing shape scores, SPSS models additionally included difficulty weightings for each image determined by professionals. This allowed the model to determine how much of the variability between image comes from the difficulty of an image. The TLX like questions were converted from 1-5 rating to 0-100 score and mean $/ \mathrm{Cl}$ determined, along with ICC and non-inferiority. The same outlier images excluded from analysis in segmentation testing were again excluded.

## E.4.2.2 Results

Table E.2: Simplification summary test results. This table shows how users performed when naming shapes perceived, categories interpreted, and object name between algorithms. A version allowing for difficulty rating by professionals for original images to weight the shape scores is included. It also shows how users rate each algorithm with the system usability survey, task load, and subjective rating preference or rank. Relative risk indicates how many times more likely a user is to answer incorrectly for a particular algorithm compared to the professionally generated images (algorithm 0). Lower ranking means higher preference.

|  | Mean Correct (95\% CI) |  |  |  |  |  | ICC | Relative Risk |  |
| :--- | :--- | :--- | :--- | :--- | :--- | :--- | :--- | :--- | :--- |
| Non-Inferior |  |  |  |  |  |  |  |  |  |
| Algorithms | 0 Pro | 2 LS |  | 4 Simp |  | 2 | 4 | 2 | 4 |
| Shape | $26 \%$ | $(6 \%)$ | $24 \%$ | $(8 \%)$ | $18 \%$ | $(3 \%)$ | 0.16 | 1.0 | 1.0 |
| True | True |  |  |  |  |  |  |  |  |
| Shape Wgt. | $25 \%$ | $(4 \%)$ | $23 \%$ | $(6 \%)$ | $18 \%$ | $(3 \%)$ | 0.16 | 1.2 | 1.1 |
| Category | $15 \%$ | $(3 \%)$ | $16 \%$ | $(6 \%)$ | $08 \%$ | $(4 \%)$ | 0.22 | 1.2 | 1.6 |
| Name | $12 \%$ | $(3 \%)$ | $10 \%$ | $(5 \%)$ | $04 \%$ | $(3 \%)$ | 0.26 | 1.5 | 1.7 |
| True | True |  |  |  |  |  |  |  |  |
| Usability | 50 | $(21)$ | 41 | $(25)$ | 39 | $(22)$ | 0.13 | - | - |
| Lrue | True |  |  |  |  |  |  |  |  |
| Load | 48 | $(16)$ | 49 | $(15)$ | 44 | $(15)$ | 0.38 | - | - |
| Rank | 1.6 | $(0.5)$ | 2.0 | $(0.5)$ | 2.5 | $(0.5)$ | 0.53 | - | - |
| True | True |  |  |  |  |  |  |  |  |

## Shape Mean Score



Figure E.21: Shows how users performed when listing shapes felt. Error bars indicate confidence interval.

Shape Mean Score


Figure E.22: Shows how users performed when listing shapes felt with difficulty accounted for. Error bars indicate confidence interval.

Category Mean Score


Figure E.23: Shows how users performed when listing category felt. Error bars indicate confidence interval.

## Name Mean Score



Figure E.24: Shows how users performed when listing name felt. Error bars indicate confidence interval.

Table E.3: Simplification modeling variables found significant from GEE.

| Variable | Name | Category | Shape | Shape Wgt. |
| :--- | :--- | :--- | :--- | :--- |
| Algorithm | X | X | X | X |
| Image Class |  |  |  |  |
| Algorithm Order |  |  | X | X |
| Algorithm * Image Class |  |  |  |  |
| Algorithm * Algorithm Order | X | X |  | X |
| Difficulty | - | - | - | X |

User interaction During interaction with the study participants, study staff observed some trends in user behavior. While describing shapes some users reported small domes as feeling like triangles. Some users drifted vertically away from lines as they felt long horizontal objects and lost contact, resulting in them having to return to feel the object again or becoming confused. This was not common with long vertical objects. Users rarely changed their ranking opinion after they learned what the diagrams contained and asked which were better at conveying the content. Three subjects performed considerably better than the other subjects. Interestingly, they reported minor prior use of
tactile graphics compared to the other subjects that reported no use of tactile graphics. Table E. 4 shows the difference in shape descriptions between those with some (novice) and those with no experience (fundamental awareness) with tactile graphics. Most users needed to be frequently reminded to explain what they are feeling as they explore the diagrams.

Table E.4: Shows difference in mean shape scores between users with some experience and those with no experience. (Proficiency scale [69])

| Algorithm | 0 | 2 | 4 |
| :--- | :--- | :--- | :--- |
| Novice | $41 \%$ | $30 \%$ | $21 \%$ |
| Fundamental Awareness | $19 \%$ | $19 \%$ | $15 \%$ |

User comments While attempting to describe images, several participants struggled to move forward in exploring some images, citing that they felt they needed to know more information about the image before they could describe it. This happened regardless of the process used to make the tactile diagram. One user suggested a light background grid to help with orientation of the hand on the diagram. Several users commented that sometimes diagrams created by algorithm 4 (Simp) had too little information because too many lines were removed. However, they sometimes reported with the same algorithm that there was too much information leaving it confusing. Users did not comment on the amount of information in the LS and Pro images.

Some users commented that they ranked algorithm 2 (LS) the best because it felt smoother from all the noise making the paper feel velvety. Following the first sentence you may want to add: this suggests that the ranking that participants performed may have been based on tactile aesthetics rather than the usefulness of conveying information. The ranking was designed to correct for this by telling participants what the content of an image was after they first felt the tactile diagrams produced by the different methods. However, several subjects also commented that once they picked their favorite, knowing what the images were and being asked to pick the best to convey the contents,
did not sway their option. Some users commented that the Pro images were tactually pleasing because of the smoother lines but that that did not change their ability to convey information.

Professional comments After user testing professionals viewed several examples of the simplified images ranging from the best to the worst to help researchers analysis the results. Overall they indicated that they thought performance with the simplified images should have been better if participants focused on learning something about the diagram they are feeling. They said that the poorer subject performance may have been due to distraction from too many questions and lack of experience with tactile graphics.

## E.4.2.3 Discussion

Overall, performance with tactile diagrams made by both computer algorithms was non-inferior compared to performance with professionally made diagrams. When considering the risk of incorrectly describing shape, both algorithms performed similarly to the professionally made diagrams. However, cognitive load was significantly greater when participants used the tactile diagrams generated by algorithm 4 (Simp) as compared to algorithm 2 (LS). This is consistent with the rankings made when users compared the different tactile diagrams of an image produced by the different algorithms side by side, they generally ranked diagrams made by algorithm 0 (Pros) as their favorite with algorithm 2 (LS) as their second preference. Although rank performance my have less to do with actual performance and more to do with aesthetics of the diagram based on user comments. Based on user's varied descriptions of too little or too much content for image for algorithm 4 (Simp), a future design consideration may be to add/remove details based on the learning objective. Additionally, since the diagrams produced by algorithm 2 (LS) contain more information than those produce by algorithm 4 (Simp), future implementations may use both together in a series of diagrams. This could help users build from general to more detailed images when learning about the contents of a diagram, as is sometimes
done by TVIs. Algorithm order seemed significant for most models and showed performance was better for the first block then users progressively performed worse for each block. This likely indicates cognitive fatigue not only requiring more breaks but also fewer questions asked of the participants.

## Descriptions

Category Many users first considered categorizing objects before identifying shapes, attempting to match the image to categories in their own personal order of preference; for example, one subject would determine if it was a type of car first, then a type of animal, then bug, etc.

Shape Once the subjects felt comfortable with the category they assigned they then searched for and identified only shapes that confirmed their hypothesis. For some users, if their mental models broke down because they could not confirm their hypothesis or they exhausted their list of standard categories they would simply say they could not explain the shapes and express a desire to end the trial. With a simple reminder of the prompt by the researcher the subject would then provide more shape description. Other users with more experience would attempt to continue identifying shapes and build a new model. This may indicate there is another low level component besides shape that is interfering with the mental development of the tactile graphics into a complete object. Based on user comments there may also be a dependence on top-down processing from users that lost their vision later in life. For these users lower performance may be due to residual dependence on the way vision has a mix of top-down and bottom-up processing to form schema and object understanding from images. Slightly more experienced users seemed to not be as limited by this dependance as they would attempt to build new models when necessary. This cannot however be confirmed with this small sample group.

Name After discussion shapes, some users were reluctant to name objects for fear of being wrong, some simply said it was too hard, and others expressed having fun guessing the objects. Researchers reassured subjects, explaining there is no wrong answer as all answers help us improve future diagram creation. All included subjects provided names for objects even if they were unsure of their answers.

Ranking When users compared images side by side they generally ranked the professionally made diagrams as their favorite with algorithm 2 as their second preference. Some users preferred the appearance of thicker lines from noise in algorithm 2 over the others while others considered it to be cumbersome and noisy. Users were split between saying algorithm 4 had too much or too little content that cluttered image. Many users also indicated that they felt algorithm 2 and 4 could be best presented together for users to gather different information and synthesize them together.

Future Work Observing the performance difference between users with only moderately more experience and those with no experience hits that experience plays a significant role that needs to be explored or at least accounted for in analysis. Since the intended market for these algorithms is people that use tactile graphics regularly it makes sense to perform an additional experiment with only subjects having advanced to expert experience with tactile graphics. Such an experiment would also benefit from primary focus on shape descriptions since it is the most important component in building a mental model of a tactile diagram and other questions seemed to distract subjects. A reduced image set focused on images that would expect to achieve mean correct shape descriptions near 50\% for professional diagrams, would help prevent floor/ceiling effects in future testing.

Summary Users received lower scores for algorithm 0 (Pro) and algorithm 2 (LS) than in user testing of the different segmentation algorithms. It was expected that name and
category scores would be lower because users were asked to focus on shape descriptions allowing for less cognitive capacity spent on name/category. Lower overall scores are suspected to be lower either because the task load questions were distracting or users had less experience with tactile graphs than previous subjects. Many users tried to rush through describing the images to answer the task load questions questions even when prompted to provide thoughtful shape answers. Throughout the experiment and evident in initial demographic details collection there was a range of user experience with tactile graphics but more users had limited to no experience. Ideally these graphics would work universally regardless of experience however experience may play a large part in performance.

Limitations Including difficulty of images allowed SPSS to account for the influence difficulty had on scores and handle outliers accordingly. The inclusion of difficulty only marginally affected Pro and LS while not affecting Simp scores. With more data and a larger difference this might have indicated Simp was less affected by difficulty. However, due to a small data set, small difference, low overall scores, and probable floor affects this cannot be definitively determined.

Another potential limitation of the study could be determined by the scores obtained for all algorithms. General scores were low with large positive skews, indicating wide spread floor effects making the results questionable because floor effects tend to mask the real location of mean scores. The experiment is likely too hard for subjects and needs to be improved.

## E.4.3 User Testing 2

A second study assessing user performance with the segmentation and simplification algorithms was performed to eliminate the effects of varying user experience with tactile graphics and control users? Focus on a single question. The motivation for this ex-
periment is that overall performance, and relative performance and preference, seemed to strongly depend on the experience of the participant with tactile graphics. As the use of this automatic diagram generation method is meant for those who have learned how to explore tactile diagrams and can use them well, we focus this study on this subpopulation of individuals who are BVI. Inexperienced users previously performed poorly on the professionally created tactile diagrams, as well as the computer generated diagrams. Flooring effects possibly effected the comparison between methods in previous tests which lead to only a subset of images being selected for this test. In addition, to relieve distractions, we have chosen to focus on shape descriptions. Algorithm 5 (SimpT) was included in this test to determine if difference in texture salience impacted results.

## E.4.3.1 Method

The experimental procedure was predominantly the same as in the first simplification user study. However, in the experimental component where participants were to describe the diagrams, they were only asked to describe the shapes in the diagram. Users were not asked about category or names of objects. Four algorithms were tested in this study: algorithm 0 (Pros), algorithm 2 LS), algorithm 4 (Simp), and algorithm 5 (SimpT). Algorithm 5 used the simplification method described earlier in the chapter combined with texturizing that produced full height textures with no buffer between texture and edge lines. The modified NASA task load index was also only performed at the end of each algorithm. When ranking images subjects were immediately informed of the image contents and asked which image best helps the user understand the object. The 8 image groups with highest consistency and near 30\% accuracy based on professional mean shape description score in previous testing were used for further shape description testing and between algorithm ranking. We expecting that having more experienced users would increase mean professional shape description scores to about 50\% to avoid ceiling and floor affects. Previously each image within variant groups were not repeated between the
three presented algorithms because there were three image variants per group. Since an additional algorithm was presented some images were necessary randomly selected to be repeated. Data was analyzed in the same manor as previous testing.

Participants Different test subjects were used than those who were in the segmentation methods assessment study and previous simplification study. Participants were required to have had experience with tactile diagrams in middle or high school and continued use within the last 5 years. Five subjects participated. Participants were over the age of 18 and blind or visually impaired. Blindness ranged from congenitally blind to adventitiously blind, and from totally blind to light sensitivity. Participants did not have multiple disabilities. All subjects indicated that they use tactile diagrams and braille regularly in their lives.

## E.4.3.2 Results

When statistically compared, all algorithms were non-inferior when it came to user shape description. However, users performed the best with professionally made diagrams, algorithm 0 (pros), with both methods using simplification tied for second. The rater reliability (intraclass correlation coefficient) was higher than in previous testing for shape descriptions. Performance between simplification algorithms was not statistically significant and performance between algorithm 0 (Pros) and algorithm 5 (SimpT) were not statistically significant. Relative risk showed that users were about as likely to answer incorrectly for the computer generated diagrams as they were for professionally made diagrams. Algorithm 2 (LS) had lower mental load than the other methods but due to large variance this is not significant. Rank results were unable to show statistical significance between methods. Figure E. 26 and Table E. 7 show the results of considering only the 8 images used in this study for the advanced users in this study and the same 8 images for novice and intermediate users in the last study.

Table E.5: Simplification summary test results. This table shows how users performed when naming shapes perceived. It also shows how users rated each algorithm in terms of the task load with the modified NASA task load index and the subjective rating preference or rank. Relative risk indicates how many times more likely a user is to answer incorrectly for a particular algorithm compared to the professionally generated images (algorithm 0). Lower ranking means higher preference.

|  | Mean Correct (95\% CI) |  |  |  |  |  | ICC | Relative Risk |  |  |  |
| :--- | :--- | :--- | :--- | :--- | :--- | :--- | :--- | :--- | :--- | :--- | :--- |
| Algorithms | 0 Pro | 2 LS |  | 4 Simp | 5 SimpT |  | 2 | 4 | 5 |  |  |
| Shape | $84 \%$ | $(7 \%)$ | $68 \%$ | $(7 \%)$ | $77 \%$ | $(9 \%)$ | $78 \%$ | $(8 \%)$ | 0.37 | 1.08 | 0.98 |
| Load | 83 | $(21)$ | 73 | $(18)$ | 80 | $(13)$ | 80 | $(12)$ | 0.47 | - | - |
| Rank | 2.1 | $(1.0)$ | 2.8 | $(1.2)$ | 2.7 | $(0.9)$ | 2.5 | $(0.8)$ | 0.05 | - | - |

## Shape Mean Score



Figure E.25: Shows how users performed when listing shapes felt. Error bars indicate confidence interval.

Table E.6: Simplification modeling variables found significant from GEE.

| Variable | Shape |
| :--- | :--- |
| Algorithm | X |
| Image Class |  |
| Algorithm Order | X |
| Algorithm * Image Class |  |
| Algorithm * Algorithm Order | X |

Table E.7: Shows difference in mean shape scores between users with different experience levels. (Proficiency scale [69])

| Algorithm | 0 | 2 | 4 | 5 |
| :--- | :--- | :--- | :--- | :--- |
| Advanced - Expert | $85 \%$ | $68 \%$ | $77 \%$ | $77 \%$ |
| Novice - Intermediate | $31 \%$ | $30 \%$ | $28 \%$ | - |
| Fundamental Awareness | $24 \%$ | $16 \%$ | $15 \%$ | - |



Figure E.26: Shows difference in mean shape scores between users with different experience levels. Error bars show 95\% CI. (Proficiency scale [69])

Table E.8: Simplification modeling variables found significant from GEE when comparing experience level.

| Variable | Shape |
| :--- | :--- |
| Algorithm | X |
| Image Class |  |
| Algorithm Order |  |
| Algorithm * Image Class |  |
| Algorithm * Algorithm Order |  |
| Experience | X |

User interaction Users in this study required almost no time to begin working with the images. They did not need to be reminded to explain what they were perceiving as they explored the diagrams. Two of the participants mentioned they could have probably guess many of the object names but did not since they were instructed to focus on shape. The rest of the participants said they would not have known what the objects were just the clarity of the shapes.

User comments Users found the spines in simplified diagrams to be less distracting to image understanding than level set's noise for most images. Some subjects commented that they liked the velvet feel of the noise in level set images but it did not help in understand images and they preferred the simplification images for gathered information. Most subjects liked the more well defined lines in the professional diagrams and to a lesser extent those in the simplified diagrams. While ranking images and discussing the two versions of simplification they often cited for a particular image preferring either the damped texture or the full texture depending on specific image context. For example, one subject preferred the full texture for the phases of the moon while preferring damped texture for the rocket ship. When ranking images between the two simplification methods that were identical because the particular image lacked texture the subjects repeatedly described the images as being considerably different.

## E.4.3.3 Discussion

This experiment showed clearer more reliable results than earlier testing as evident by the fact that data distributions were no longer skewed, data was not in jeopardy of ceiling or floor effects, and participants were better engaged in the task. This experiment confirmed that all versions of computer generated graphics were non-inferior and nearly the same relative risk as diagrams made by professionals. Shape description and task load rater reliability was higher compared to previous testing indicating improved testing methods. Both simplification methods improved upon segmentation alone seeming to
reduce the performance gap between level set and professional diagrams by over half. Level set segmentation alone was significantly different from professional diagrams but algorithm 5 (SimpT) was no longer significantly different. Both simplification methods proved to be great methods for creating tactile graphics automatically as they performed well and model the available body of knowledge for the creation of tactile graphics.

Noticing that performance and task load are lower for algorithm 2 (LS) might indicate that users gave up exploring the diagrams before describing all they shapes they encountered. They reported experiencing lower load because they did not push themselves as hard to explore these diagram, potentially even because they felt extra comfortable with these diagrams due to the pleasing texture. Future testing could include tracking the amount of time spent on each diagram to further explore this possibility.

Observing the marginal differences in performance between algorithms for each experience level, might hit at the possibility that different computer generated methods are better for users at different places in the learning process. It might also hit that: early users require the structure of professional diagrams to start learning how to use tactile graphics; then by their intermediate state they are at an inflection point where they can continue their learning process with a wider range of graphic sources; then by expert stage they have tuned their abilities to using professional diagrams making computer generated graphics require additional thought. Further testing would be required to explore the learning process involved with the use of tactile graphics to definitively determine is this observation is noise or actual phenomenon.

## E.4.3.4 Limitations

Limitations included: small subject sample size due to significant difficulty in recruiting participants old enough to convey their experience exploring diagrams that still regularly use tactile diagrams; and limited number of diagrams due to our desire to avoid floor and
ceiling effects.

## Chapter F: Postface

## F. 1 Conclusions

## F.1.1 Segmentation Development

"To explore/modify/develop a variety of automated techniques to segment an image/drawing into a set of non-fragmented lines/curves that are effective for tactile information processing. Effective will be defined as the ability to convey information about the shape(s) of objects and their parts and/or the identification of objects and their parts, as well as identification of the main spatial relationships between objects and/or object parts in the graphic." [B.2]

Progress Several base segmentation algorithms along with their constituent components and a wide range of features were explored. Our intent for segmentation was not traditional visual segmentation but segmentation for the conversion of images to tactile diagrams. The current state of segmentation algorithms are optimized for traditional visual segmentation. Appropriate improvements to the k-means algorithm and extensive improvements to the level set algorithm were undertaken to develop a segmentation model that would work like professional diagram makers. Next the algorithm improvements were extensively tested in an objective manor to optimize the algorithms for automatic conversion of visual images to tactile graphics. Improvements included expansion to new feature spaces, exploration of newer feature space measurements, and alterations to the governing equations.

Advancements Past work focused on making tactile graphics from visual images with edge detection and some segmentation for BVI subjects but they did not base their method on how TVIs make graphics $[9,70]$. Their edge based methods are not able to receive texture which requires closed regions. Their k-means based methods look good but due to the differences in visual vs tactile diagram perception a comparison to pro-
fessionally made diagrams indicating tactually perceptual diagrams was needed. Other work focused on general TVI procedures but again did not model their optimized outputs to match TVI generated tactile diagrams [6, 10]. They also focused on math diagrams and plots with well defined parts. Their methods worked great for math diagrams but they would not be able to work with photos and natural scenes. We went further to model not only the procedure based on professionals but also optimized our segmentation algorithm to be the closest to professionally made diagrams and optimized for speed in future real-time uses.

Exploration Several classes of segmentation algorithms were identified to be unsuitable for making tactile graphics in a reasonable time. The k-means algorithm was shown to not work as well in producing tactile graphics, likely because it can only operate on continuous grouping features. Past research focused around k-mean and though not the same as our was similar. Visually the images look nice but they leave inconsistencies that make tactile graphics hard to interpret. We identified our new distance to pro measurement in PR space as being unable to account for P vs R balance and select the best parameters as intended. PRI was the best at digitally comparing computer generated tactile graphics with those made by professionals.

## F.1.2 Segmentation User Testing

"To quantitatively compare, for users who are blind or visually impaired and determining by touch alone (those with residual vision will be blindfolded), the effectiveness and ease of use of the two best automated visual-to-tactile conversion methods developed to visual-to-tactile conversions made "by hand" by expert tactile graphic makers. Effectiveness is defined as above and will be assessed by asking questions to the user about the diagram. Ease of use will be defined by the use of the System Usability Scale." [B.2]

Progress Unlike other tests, we developed a testing strategy that incorporated testing computer generate graphics against professionally made graphics. This allowed us to ac-
curately determine if the generated graphics were able to convey the required information to understand the graphics with similar performance as professionally made diagrams. Our testing method, intentionally stress tests diagram performance by having users describe the diagrams without text labels nor teacher assistance to mimic their independent use. The protocol also assessed both objective measures such as performance with subjective measures of usability. Our developed level set method performed above $75 \%$ of professionally made diagram performance and the gPb-owt-ucm performed above $55 \%$ of professional diagrams.

Advancements Past work focused on making tactile graphics from visual images with edge detection and some segmentation but only reported success from testing of their work between sighted and BVI subjects with no comparison to professional diagrams [ 9,70 ]. They performed user testing that focused on questions that required the subject know about the object depicted by locating parts of the object or by naming the object from a list of choices. Other work focused on general TVI procedures but did not perform user testing $[6,10]$. We went further to not only model our algorithm to closely match TVI made tactile graphics but also performed user testing with both professional images and images generated from our algorithm. This allowed us to establish a baseline for determining a target performance goal. Our testing went further to study if subjects could use our diagrams to perceive low level shapes necessary for building a mental model of an object they may never have encountered before.

## F.1.3 Simplification Development

"To explore/modify/develop a variety of automated techniques to simplify an image/drawing into a set of non-fragmented lines/curves that are effective for tactile information processing. Effective will be defined as the ability to convey information about the shape(s) of objects and their parts and/or the identification of objects and their parts, as well as identification of the main spatial relationships between objects and/or object parts in the graphic. " [B.2]

Progress Diagram simplification is an important step in tactile diagram simplification [71]. We developed models of the major steps professionals take to illustrate tactile graphics. From our model, we developed an algorithm for the automatic creation and simplification of tactile graphics. The model included the key steps professionals take to outline lines, segment images, clean unnecessary regions, and apply textures to segments. TVIs tend to make iterative changes to diagrams then inspect their diagrams and continue to make changes. Our model does not need to work iteratively and goes through the steps only once. Our model not only explored cues from local to global but also cues based in simple calculation to those requiring non linear processing and even Google image processing. Cues from all levels along with some quadratic and interactions terms were refined to a small set to improve run-time without losing accuracy.

Advancements Past work has focused on edge detection with some newer work on segmentation $[9,10,70]$. Though they mention simplification in their work it has a different meaning compared to our simplification. In some work simplification is a pre-processing technique to reduce the number of colors and thicken lines for printing [6, 10]. In other work the term simplification is all encompassing term for the conversion of visual images to tactile graphics [9, 10, 70]. I past work with simplification their operations are applied without discrimination to all images equally. We took simplification to a new level with major developments that greatly alter the segmented images. We did not make the same changes to all images but rather modeled the changes based on cues within the images themselves; so that low to high level cues extracted from the images could be used to predict the simplification decisions made by professionals. This change allowed the method to adaptivity make changes that were best for those images. We both added and subtracted content from segmented images in order to make automatic computer generated tactile graphics more understandable users.

Exploration Throughout the modeling process we examined and identified many potential low to high level cues that might influence the way professionals make tactile graphics. In the process of automatic feature selection we were able to identify many cues that were not applicable to improving tactile graphic usability.

## F.1.4 Simplification User Testing

"To quantitatively compare, for users who are blind or visually impaired and determining by touch alone (those with residual vision will be blindfolded), the effectiveness and ease of use of the two best automated visual-to-tactile conversion methods developed to visual-to-tactile conversions made "by hand" by expert tactile graphic makers. Effectiveness is defined as above and will be assessed by asking questions to the user about the diagram. Ease of use will be defined by the use of the System Usability Scale." [B.2]

Progress We have developed a testing protocol for comparing computer generated graphics to those made by professionals. Our algorithm for converting visual images to tactile diagrams with simplification achieved $90 \%$ of the accuracy of professionally made diagrams for people that used tactile graphics frequently. The difference in performance among methods was far out shadowed by much larger differences between experience levels. Many new tactile graphics users commented that they loved segmentation alone because it felt nice where as the expert users commented they like the feel of segmentation along but segmentation with simplification conveys information content much better. This is a considerable step towards improving access to tactile graphics for people who are blind or visually impaired. The process of conversion, without speed boosts that can be achieved outside of Matlab, and while extracting all cues explored not just those required by the model, takes an average 89 seconds for images averaging $1140 \times 1500$ pixels.

## F.1.5 Limitations

The use of SUS and TLX-like questions was designed to help indicate subjectively cognitive fatigue and user experience. These measures were not able to indicate any significant differences between professional and computer diagrams. Future user testing could benefit from measuring how long users spend exploring diagrams to answer the questions as an objective metric. Taking longer to explore diagrams similar in difficulty with different algorithms is likely to reflect the difficulty in using tactile diagrams made from a given algorithm. However, this metric is likely to interact with the correctness of the actual answer, as users may give up early if they feel the diagram is too difficult to explore. Thus, any analysis should consider both together. Even so, such a measure might be problematic: if users are told they are being timed, they might try to explore as fast as possible and may not explore as much of the diagram. This may also effect the correctness of the answer they give. On the other hand, if users are not told they are being timed they may get caught up in the description of little points, distracted by ambient stimuli or try to make conversation unrelated to the tactile diagrams.] timing subjects would also prevent the study staff from probing the participant to gain insights into what the users are thinking and experiencing.

Many images in our image set have similar low-level information/cues but based on the object itself requires more or less detail to build a mental model of what it is. Future improvements to these models is likely to come from better understanding of when an object/object scene needs more or less detail. This may come from the context the image is in, such as a teacher giving students their first look at a turtle versus another teacher explaining the different types of turtles. It may also come from a better cognitive understanding of the object in an image.

## F.1.6 Population Differences

Based on the difference in results between advanced-expert diagram users and novices, future automated graphics validation testing should start with advanced-expert users to get a clear picture of performance for the main audience (people that use graphics regularly and are able to use them independently). Then, if the advancements being tested are to help people learn how to use tactile graphics, they can be tested with novice users. Advanced users were able to immediately start explaining what they were experiencing in the graphics. Whereas, the new users, needed to be reminded to talk and explain what they were feeling. The new users often complained that they needed to know what the diagram is before they could explain what they were feeling. The advanced users were already proficient as using tactile graphics in a bottom-up fashion to build the diagram into a mental model of the object. Advanced users were better able to communicate their experience with less fatigue generating clearer data.

## F. 2 Future work

Although there are guidelines for teachers for developing tactile graphics there is limited to no research on how students best learn to use tactile graphics and how they learn to build mental models. Some users try to balance a top-down/bottom-up methods as they would with vision, whereas others over time have learned to focus on bottom-up methods and progressively integrate past experience while they continue to explore a diagram. The lack of knowledge in the literature means that in experimental studies participants cannot be instructed as to the best way to perform tactile diagram exploration; they instead learn this process primarily through trial and error or teacher intuition (when a TVI is available). This produces a high degree of variation in the results, particularly for inexperienced users. Some users have developed their own sustainable methods for exploring graphics. This reveals a place for future research for the development of teaching
methods that help novice users improve their bottom-up processing of tactile graphics.
To further support improved teaching methods, it is possible that some of this insight might come from understanding how users explore diagrams and how this may change based on the cognitive context given. Also understanding how users explore diagrams could help address the gap between novices and experts as not all individuals who become blind have access to TVIs to help them learn about graphics. Research could provide instructions or an automated assistant to help instruct inexperienced users.

To improve the automatic creation of tactile graphics any further, deeper research is needed to understand how a user interprets the diagrams so that each diagram can be tailored to the user based on the contents of the diagram. Insights into the learning process will likely influence the segmentation and simplification models as well as reveal new directions for further improvements.

Another issue in the interpretation of tactile graphics seems to be the effect of the medium itself. In the performed studies, users seemed to feel subtle differences in the production of printed tactile graphics that influenced performance, particularly for novice users. To begin with, pages of micro-capsule paper from different production batches feel different on their own. This difference is exacerbated in the printing process by the variability in the response of the microcapsules at different locations on the paper causing the lines and textures to feel different to users. Before any further improvements in testing protocol and algorithm development can be researched, a more consistent testing medium is recommended. Considering this seems to only be a problem for novices, this might be solved with automated instruction/guidance to train exploration and help users learn to use tactile graphics.

Including more images made by more professionals may help improve the model by reducing over fitting and make the model work border also reducing the change of too much or too little information in some images. We expect this to only be a marginal refinement.

Some users commented when asked to rank the image produced by different methods that they liked the ones with less information for building a framework and orienting themselves then moving to the diagrams with more content to learn about the details of the objects. Testing to see if users better understand images when they are given multiple tactile graphics of the same image made at different detail levels will help us understand if users perform better by being forced to build a mental model in stages. Exploring how well students learn new concepts from these diagrams (vs pros) in classroom not just ability to explain what they are feeling and parts of objects will help us ensure the diagrams are assisting in learning and give insights into the learning process.

For future use by professionals or students at home a more user friendly method is needed with fewer parameters to work with. Developing a model that simplifies the large number of cues we have, to a single option, where the only input is a desired level of detail, will help teachers more easily use it and also improve future studies. Taking this improvement further, if presenting multiple levels of detail is shown to help students learn to build mental models better, then developing a model that produces these multiple levels automatically would be beneficial. Developing this is only possible if first a model based on the object itself is developed; one that interprets whether more or less detail is needed based on the context of the image to improve upon the current images that sometimes have too little or too much information. Developing a model for interpreting text descriptions with natural language processing to find context and decide on a level of detail will help users better understand the graphics and further improve user autonomy.

Further research on how users interpret and learn to interpret tactile graphics will help improve our model. Developing a model that incorporates the meaning of a diagram will assist in deciding the correct level of detail to be displayed to users. Combining the use of Google image processing to automatically determine graphics labels will allow for the development of automatic tactile graphics where there are no labels. Using Google image processing to find object names within an image should also help determine the level of
detail needed, such as when diagrams are labeled as cartoons by Google indicating they may need a different path for processing than natural images.

Appendix G: Presentation Slides

## VCU Rehabilitation <br> Technology and Haptics Laboratory <br> Automatic Image Processing and Conversion to Tactile Graphics

## Tyler Ferro

## 

## Contents

- Introduction
- Segmentation
- Simplification
- Closing


## 

## Introduction

- Population Needs
- Tactile Graphics
- Manual Methods
- Automatic Methods
- Objectives

$\qquad$

Population Locations

$\qquad$
$\qquad$
$\qquad$
$\qquad$
$\qquad$

Target Use $\qquad$

- No knowledge of image $\qquad$
- Independent use
- Fully automated
- Ease TVI overload
$\qquad$
$\qquad$
$\qquad$
$\qquad$
$\qquad$

$\qquad$

$\qquad$
$\qquad$
$\qquad$
$\qquad$
$\qquad$

Commercial Software

$\qquad$

## Past Research



## 

Past Research

$\qquad$

## Past Research

$\qquad$

- "Locate the tail fin of the space shuttle."
- "Is the shuttle landing from left to right, or right to left?"
$\qquad$
$\qquad$
$\qquad$
$\qquad$
$\qquad$

[^0]$\qquad$

## Objectives

- To develop a variety of automated techniques to segment images for tactile graphics
- To compare segmentation performance from developed techniques with images from professionals
- To develop a variety of automated techniques to simplify images for tactile graphics
- To compare simplification performance from developed techniques with images from professionals



## Segmentation

- Decisions
- Implementations
- Optimization
- Testing
- Summary


## 

## Decisions


$\qquad$
$\qquad$

## Feature Space

- Energy (Gradient) - Roberts, Prewitt, Sobel, Kirsch, Laplacian of Gaussian, susAN, Phase Congruency, Monogenic Signal, Oriented Energies, Oriented Gradient of Histograms, Probability of boundary
- Orientation - Roberts, Prewitt, Sobel, Kirsch, Phase Congruency, Monogenic Signal, Oriented Energies, Oriented Gradient of Histograms
- Phase - Phase Congruency, Monogenic Signal



## Feature Space

- Texture - texture gradient, texture classification
- Distance - Euclid, Dihedral, Normalized, Mahalanobis
- Spectral - Eigen values of affinity matrix

$\qquad$
$\qquad$
$\qquad$


Decisions

- Edge Detection
- Graphing
- Clustering
- Contour Detection
- Differential Equations
$\nabla$
- K-Means
gPb-owt-ucm
Level Set


## K-Mean

- Feature space
- Iterative
- Distances


## 

## Segmentation

K-Means, Color Space Distance


Standardized


Standardized (X,Y)


Mahalanobis
gPb-owt-ucm

- Berkeley
- Textons
- Multiscale Pb
- Spectral Pb
- Global Pb
- Oriented watershed transform
- Ultrametric Contour Map
- Extensively explored


## 

$\qquad$

Level Set


Level Set ( $\Psi, \phi$ )

 $\underset{23}{ } \operatorname{Segmentation}^{23}$

Differentials
$\Psi(x, y)= \pm d(x, y)$ $\qquad$
$\Psi_{t}+F(x(t))|\nabla \Psi|=0$
$d \Psi=-F(x(t))|\nabla \Psi| d t$
$F(x(t) ; t)=\alpha\left(P_{f}(I)-P_{b}(I)\right)+(1-\alpha)(c+\kappa(x(t) ; t)) g(\nabla I)$
$\kappa(x(t) ; t)=\operatorname{div}\left(\frac{\nabla \Psi}{|\nabla \Psi|}\right)$
$\frac{\partial \Psi}{\partial t}=\delta_{\epsilon}(\Psi)\left(\mu \kappa-\lambda_{1}\left(I-\mu_{1}\right)^{2}+\lambda_{2}\left(I-\mu_{2}\right)^{2}\right)$


## Weighted Sum

| $\phi^{\prime}=w_{1} \cdot \omega(\phi)+w_{2} \cdot \gamma(\Psi) \cdot \lambda \cdot \phi+\varepsilon$ |  |
| :--- | :--- |
| $\phi^{\prime}=w_{1} \cdot \omega(\phi)+w_{2} \cdot(c+\gamma(\Psi)) \cdot \lambda \cdot \phi+\varepsilon$ |  |
| $\phi^{\prime}=w_{1} \cdot \omega(\phi)+w_{2} \cdot \lambda \cdot \phi+w_{3} \cdot \gamma(\Psi) \cdot \phi+\varepsilon$ | [This work] |

[48] $\qquad$
$\qquad$


Level Set


[^1]
$\qquad$
$\qquad$
$\qquad$
$\qquad$
$\qquad$
$\qquad$
$\qquad$

Image Space ( $\phi$ )

$\qquad$
$\qquad$
$\qquad$
$\qquad$
$\qquad$
$\qquad$
$\qquad$

Grouping Space ( $\omega$ )


Segmentation
29

Grouping Space Distance $(\omega(\phi))$

$\qquad$
$\qquad$
$\qquad$
$\qquad$
$\qquad$
$\qquad$

$\qquad$
$\qquad$
$\qquad$
$\qquad$
$\qquad$
$\qquad$

Damping Space $(\gamma(\Psi))$



Logical Decisions

- Initialize
- Merge
- Split
- Stop
- Clean



## Parameter Selection

- Match professionals
- Time
- Consistency

$\qquad$

Parameter Selection


$\qquad$
$\qquad$

$\qquad$
$\qquad$
$\qquad$
$\qquad$
$\qquad$
$\qquad$
$\qquad$

$\qquad$
$\qquad$
$\qquad$
$\qquad$
$\qquad$

$\qquad$
$\qquad$
$\qquad$
$\qquad$
$\qquad$
$\qquad$
$\qquad$

$\qquad$
$\qquad$
$\qquad$
$\qquad$
$\qquad$
$\qquad$


## Optimization

- Select best of best (per algorithm)
- Select best 2 algorithms
- gPb-owt-ucm (40 sec*)
- Level Set (20 sec)



## Segmentation

## Optimization

Metric Overall K-Means Level Sets gPb
Probabilistic Rand Index (PRI) $1.93:: 1.57$ 2.73:: $2.001 .79:: 1.40$ 1.81 :: 2.00
Regression (Reg-PRI, RG) 2.08 :: $1.71 \quad 2.54$ :: $2.00 \quad 2.15$ :: $1.80 \quad 1.32$ :: 1.00
F-Score $\quad 2.16:: 2.14 \quad 3.00$ :: 3.00 2.15 :: $2.20 \quad 1.35$ :: 1.00

| Loomis Distance | $2.23:: 2.14$ | $3.59::$ | 4.00 | $2.14::$ |
| :--- | :--- | :--- | :--- | :--- |
| 2.00 | $1.32::$ | 1.00 |  |  |

(Mean :. Median)

$\qquad$
$\qquad$

## Optimization

| Algorithm Set | Mean | Median |
| :--- | :--- | :--- |
| gPb-owt-ucm | 3.1 | 2.0 |
| Levelset Init | 3.3 | 2.0 |
| Levelset Noise | 3.2 | 3.0 |
| kMean Base | 3.2 | 3.0 |
| Levelset MS | 4.4 | 5.0 |
| LevelSet Base | 5.1 | 5.0 |
| Levelset Weight | 5.2 | 6.0 |



## Segmentation

Testing

- Images - 81 (+9 training)
- Methods - Pro, gPb, LS
- Participants - 7
- Trials - Counterbalanced blocks
- Environment - Blindfolded
- Analysis - GEE
- Questions - Describe shapes


Analysis

| Variable | Shape |
| :--- | :--- |
| Algorithm | X |
| Image Class | X |
| Algorithm Order |  |
| Algorithm * Image Class <br> Algorithm * Algorithm Order | X |

[^2]$\qquad$

$\qquad$
$\qquad$


## Segmentation

Observations

$\qquad$
$\qquad$
$\qquad$
$\qquad$
$\qquad$
$\qquad$
 $\qquad$

Segmentation Summary

- Level set - best
- Simplification needed

$\qquad$
$\qquad$
$\qquad$
$\qquad$
圈 $\qquad$


Human Basis $\qquad$
$\qquad$
$\qquad$



| Simplification |
| :---: |
| 50 |

Decisions $\qquad$


Preparation


Preparation




Feature Selection

| Method | Adj R 2 | Outliers | Edge Outliers |
| :--- | ---: | ---: | ---: |
| Professionals to each other | 0.30 | - | - |
| PCA, Normalized | 0.84 | $7.48 \%$ | $9.5 \%$ |
| PCA, Standardized, Pre-regressed | 0.94 | $30.90 \%$ | $57.9 \%$ |
| PCA, Standardized, Trimmed | 0.97 | $28.21 \%$ | $56.4 \%$ |
| LASSO, Normalized | 0.83 | $8.89 \%$ | $0.2 \%$ |


Simplification

Feature Selection

| Method | Adj $\mathrm{R}^{2}$ | Outliers | Cues | Terms | Edge Outliers |
| :---: | :---: | :---: | :---: | :---: | :---: |
| Professionals to each other | 0.30 | - | - | - | - |
| PCA, Normalized | 0.84 | 7.48\% | 31 | 105 | 9.5\% |
| PCA, Standardized, Pre-regressed | 0.94 | 30.90\% | 20 | 85 | 57.9\% |
| PCA, Standardized, Trimmed | 0.97 | 28.21\% | 28 | 102 | 56.4\% |
| LASSO, Normalized | 0.83 | 8.89\% | 14 | 37 | 0.2\% |
| PLS, Normalized | 0.65 | - | 225 | 10 | - |
|  |  |  |  |  | Simplification |

## Feature Selection

$\qquad$

- Cue Select
- Cue Reduce


[^3] Simplification 57

Edge Modeling

$\qquad$
$\qquad$
$\qquad$
$\qquad$

Simplification

Thinning

$\qquad$
$\qquad$
$\qquad$
$\qquad$
$\qquad$
$\qquad$


$\qquad$
$\qquad$
$\qquad$
$\qquad$
$\qquad$

$\qquad$
$\qquad$
$\qquad$


$\qquad$
$\qquad$
$\qquad$
$\qquad$
$\qquad$
$\qquad$
$\qquad$

Simplified

$\qquad$
$\qquad$
$\qquad$
$\qquad$
$\qquad$

Testing 1 $\qquad$

- Images - 81 (+9 training)
- Participants - 8 (New Users)
$\qquad$
- Algorithms - Pro, LS, Simp
- Trials - Counterbalanced
- Analysis - GEE
$\qquad$


## Analysis

| Variable | Shape |
| :--- | :--- |
| Algorithm | X |
| Image Class | X |
| Algorithm Order |  |
| Algorithm * Image Class |  |
| Algorithm * Algorithm Order |  |

## 

Testing 1 Results


[^4]Midpoint Observations

| Algorithm | 0 | 2 | 4 |
| :--- | :--- | :--- | :--- |
| Novice | $41 \%$ | $30 \%$ | $21 \%$ |
| Fundamental Awareness | $19 \%$ | $19 \%$ | $15 \%$ |

$\qquad$
$\qquad$
$\qquad$
$\qquad$
$\qquad$
$\qquad$
$\qquad$

Testing 2

- Images - 81 (+9 training)
- Participants - 6 (Advanced users)
- Algorithms - Pro, LS, Simp, SimpT
- Trials - Counterbalanced
- Analysis - GEE


## 

Simplification

## Analysis

| Variable | Shape |
| :--- | :--- |
| Algorithm | X |
| Image Class | X |
| Algorithm Order |  |
| Algorithm * Image Class |  |
| Algorithm * Algorithm Order | X |

##  <br> Simplification

Testing 2 Results

$\qquad$
$\qquad$
$\qquad$
$\qquad$
$\qquad$
$\qquad$

[^5]$\qquad$

$\qquad$
$\qquad$
$\qquad$
$\qquad$
$\qquad$
$\qquad$

## Simplification Summary

- Simplification better than level set alone $\qquad$
- "[LS] feels nice and velvety but [Simp] conveys information more clearly"
$\qquad$
- Too little or too much detail balance - Context not image


## Closing

- Conclusions
- Future Considerations
- Acknowledgments


## Overall Conclusions - Objectives

- To develop a variety of automated techniques to segment images for tactile graphics
- To compare segmentation performance from developed techniques with images from professionals
- To develop a variety of automated techniques to simplify images for tactile graphics
- To compare simplification performance from developed techniques with images from professionals


## 

Novel Insights

- Detail based on object
- Need for context to improve model
- Understanding learning process


Future Considerations

- Improvements
- More data
- Level set with Berkeley DB
- User test with layers
- Short Term Research
- Consistent medium
- Model with 1 parameter (detail)
- Model with multiple layers
- Long Term Research
- How users interpret tactile diagrams
- Incorporate meaning
- Incorporate context
- Use Google image processing for labels
$\qquad$
$\qquad$
$\qquad$
$\qquad$
$\qquad$
$\qquad$
$\qquad$
- 

$\qquad$
$\qquad$
$\qquad$
$\qquad$
$\qquad$
$\qquad$
$\qquad$

[^6]
## Acknowledgments

- VCU Biomedical Engineering Dept.
$\qquad$
- Academic Advisory Committee
- Professional Advisory Committee (Janice, Kit)
- NSF IIS Grant \#1218310
- IRB Review \#HM14522
- Columbia Lighthouse for the Blind
- VA Department for the Blind and Visually Impaired
- Research Assistants (Megan, Autumn, Will)


## 

Appendix
 $\qquad$

## Appendix H: Experiments

## H. 1 Segmentation

## H.1.1 Parameter Tuning

## H.1.1.1 Parameter group values

- kMean Base (1008 combinations)
- Color Space: Gray, RGB, Luv, Lab, Gray + Texture, RGB + Texture, Luv + Texture, Lab + Texture
- Number of Clusters (k): 2, 3, 4, 5, 6, 7, 8
- Distance Measure: Squared Euclidean, Standardized Euclidean, Dihedral Angle, Euclidean + Dihedral Angle, Standardized Euclidean + Dihedral Angle, Mahalanobis
- Initialization Method: Random Sample, Uniform Distribution, Cluster (run on $10 \%$ of data first)
- LevelSet Base (432 combinations)
- Color Space: Gray, RGB, Luv, Lab, Gray + Texture, RGB + Texture, Luv + Texture, Lab + Texture
- Number of Phases (n): 1, 2, 3 ( $2^{n}=k$ number of clusters)
- Distance Measure: Squared Euclidean, Standardized Euclidean, Dihedral Angle, Euclidean + Dihedral Angle, Standardized Euclidean + Dihedral Angle, Mahalanobis
- Initialization Mask: kMean Clusters, Uniform Circles, Background Detection + Uniform Circles (use histogram to find background)
* Circle Pixel Radius: 10
* Inter-Circle Pixel Padding: 15
* Pixel Shift Between Phases: 17
- Weighted Summation
* Combination Weighting: Gradient [1], Contour Stiffness [1], Pixel Color [0.5]
* Change in Weighting over iterations:Gradient [1], Contour Stiffness [1], Pixel Color [1]
- LevelSet Weight (3375 combinations)
- Color Space: Lab
- Number of Phases (n): 1 ( $2^{n}=k$ number of clusters)
- Distance Measure: Standardized Euclidean
- Initialization Mask: Background Detection + Uniform Circles
* Circle Pixel Radius: 10
* Inter-Circle Pixel Padding: 15
* Pixel Shift Between Phases: 17
- Weighted Summation
* Combination Weighting (affects speed): Gradient [0.5,1,1.5], Contour Stiffness [0.5,1,1.5], Pixel Color [0.5, 1, 1.5]
* Change in Weighting over iterations (affects change in speed): Gradient [0.8,0.9,1,1.1,1.2], Contour Stiffness [0.8,0.9,1,1.1,1.2], Pixel Color [0.8,0.9, 1, 1.1, 1.2]
- LevelSet Init (5324 combinations) (some param re-investigated to determine if initialization could influence performance)
- Color Space: Lab
- Number of Phases (n): 1, 2 ( $2^{n}=k$ number of clusters)
- Distance Measure: Standardized Euclidean
- Initialization Mask: Uniform Circles, Background Detection + Uniform Circles
* Circle Pixel Radius: 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15
* Inter-Circle Pixel Padding: 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20
* Pixel Shift Between Phases: 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22
- Weighted Summation
* Combination Weighting: Gradient [1.5], Contour Stiffness [1.5], Pixel Color [1]
* Change in Weighting over iterations:Gradient [0.8], Contour Stiffness [0.8], Pixel Color [0.8]
- LevelSet Noise (5832 combinations)
- Color Space: Lab
- Number of Phases (n): 2 ( $2^{n}=k$ number of clusters)
- Distance Measure: Squared Euclidean
- Initialization Mask: Uniform Circles
* Circle Pixel Radius: 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15
* Inter-Circle Pixel Padding: 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20
* Pixel Shift Between Phases: 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22
- Weighted Summation
* Combination Weighting: Gradient [1.5], Contour Stiffness [1.5], Pixel Color [1]
* Change in Weighting over iterations:Gradient [0.8], Contour Stiffness [0.8], Pixel Color [0.8]
- Spatial distance plateau scale [ $\left.1^{*}, 3,5,7,9,11,13,15\right]$
- Gradient: Central difference*, Monogenic Signal, multi-scale probability of boundary, global probability of boundary
- Gradient color weight: $\left[0.5,1^{*}, 1.5\right]\left[0.5,1^{*}, 1.5\right]\left[0.5,1^{*}, 1.5\right]$ (not for mPb or gPb as those have weights built-in)
- Gradient Gaussian Smoothing: kernel size [3x3*, 6x6, 9x9], sigma [0.2, 0.4*, 0.6]
- LevelSet Merge/Stop (384 combinations)
- Color Space: Lab
- Number of Phases (n): 1, $2\left(2^{n}=k\right.$ number of clusters)
- Distance Measure: Squared Euclidean
- Initialization Mask: Uniform Circles
* Circle Pixel Radius: 5
* Inter-Circle Pixel Padding: 10
* Pixel Shift Between Phases: 12
- Weighted Summation
* Combination Weighting: Gradient [1.5], Contour Stiffness [1.5], Pixel Color [1]
* Change in Weighting over iterations:Gradient [0.8], Contour Stiffness [0.8], Pixel Color [0.8]
- Clean (remove small regions): None*, Move to Background (Largest Cluster), Move to Nearest Neighbor
- Merge: None*, Overlap (when mean/stdv overlap), Distance (when means are close enough)
- Split: None*, Force (force each spatial region to its own cluster), Histogram (if large stdv and if peaks are distinct), kMean2 (split into 2 using kMean if large stdv), kMeanR (dynamically initiated kMean)
- Determine Init number of segments: Static* (set at start), Dynamic (run several and pick based on inter/intra cluster distance)
- gPb-ucm (11 combinations)
- Color Space: RGB (Algorithm has build in texture classifier and converts RGB images to Lab)
- Threshold (k): 0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1
- Best of all other parameters such as weighted summation were found in pre-
vious work


## H.1.1.2 Regression Plots

|  |  | Wald |  |  |  |  |
| :--- | ---: | ---: | ---: | ---: | ---: | ---: |
| Term | Estimate | Std Error | ChiSquare | ChiSquare | Lower 95\% | Upper 95\% |
| Intercept | 0.5334165 | 0.0052858 | 10183.772 | $<.0001^{*}$ | 0.5230565 | 0.5437766 |
| ColorSpace_2[gray] | -0.208954 | 0.0051848 | 1624.2162 | $<.0001^{*}$ | -0.219116 | -0.198792 |
| ColorSpace_2[grayTex] | -0.04471 | 0.0046556 | 92.226165 | $<.0001^{*}$ | -0.053835 | -0.035585 |
| ColorSpace_2[lab] | 0.0426929 | 0.0048517 | 77.431043 | $<.0001^{*}$ | 0.0331836 | 0.0522021 |
| ColorSpace_2[labTex] | -0.00188 | 0.0046182 | 0.1657249 | 0.6839 | -0.010932 | 0.0071714 |
| ColorSpace_2[luv] | 0.0459969 | 0.0048869 | 88.591148 | $<.0001^{*}$ | 0.0364188 | 0.0555751 |
| ColorSpace_2[luvTex] | -0.000216 | 0.0046344 | 0.0021628 | 0.9629 | -0.009299 | 0.0088677 |
| ColorSpace_2[rgb] | 0.033932 | 0.0049173 | 47.617275 | $<.0001^{*}$ | 0.0242943 | 0.0435698 |
| kClusters[2] | 0.0610947 | 0.0048653 | 157.68444 | $<.0001^{*}$ | 0.0515589 | 0.0706305 |
| kClusters[3] | 0.0937941 | 0.0045914 | 417.30356 | $<.0001^{*}$ | 0.084795 | 0.1027932 |
| kClusters[4] | 0.070635 | 0.0044841 | 248.13464 | $<.0001^{*}$ | 0.0618463 | 0.0794237 |
| kClusters[5] | 0.04648 | 0.0044517 | 109.01314 | $<.0001^{*}$ | 0.0377548 | 0.0552052 |
| kClusters[6] | 0.0249019 | 0.0044136 | 31.832763 | $<.0001^{*}$ | 0.0162513 | 0.0335524 |
| kClusters[7] | 0.0086267 | 0.0044147 | 3.818381 | 0.0507 | -0.000026 | 0.0172794 |
| StartType[cluster] | 0.0105136 | 0.0029476 | 12.722101 | $0.0004^{*}$ | 0.0047364 | 0.0162909 |
| Startype[sample] | -0.050503 | 0.0031754 | 252.94132 | $<.0001^{*}$ | -0.056726 | -0.044279 |
| DistanceMetric[dihedral] | -0.14632 | 0.0042068 | 1209.7611 | $<.0001^{*}$ | -0.154566 | -0.138075 |
| DistanceMetric[EuclidDihedral] | -0.11781 | 0.004175 | 796.25717 | $<.0001^{*}$ | -0.125992 | -0.109627 |
| DistanceMetric[mahalanobis] | -0.221384 | 0.0042073 | 2768.7495 | $<.0001^{*}$ | -0.22963 | -0.213137 |
| DistanceMetric[nEuclidDihedral] | -0.087545 | 0.0042471 | 424.88934 | $<.0001^{*}$ | -0.095869 | -0.079221 |
| DistanceMetric[normEuclidean] | 0.0330377 | 0.0036933 | 80.018862 | $<.0001^{*}$ | 0.025799 | 0.0402764 |
| Scale | 0.2394708 | 0.0007355 | 106001.59 | $<.0001^{*}$ | 0.2380292 | 0.2409124 |

Figure H.1.1: Statistical analysis general regression of kMean results ( F measure).


Figure H.1.2: Statistical analysis general regression of kMean results ( F measure).

|  |  | Wald |  |  |  |  |
| :--- | ---: | ---: | ---: | ---: | ---: | ---: |
| Term | Estimate | Std Error | ChiSquare |  |  |  | ChiSquare | Lower 95\% |
| :--- | Upper 95\%

Figure H.1.3: Statistical analysis general regression of kMean results (PRI).


Figure H.1.4: Statistical analysis general regression of kMean results (PRI).

|  | te |  |  | Prob > ChiSquare | Lower 95\% |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| Intercept | 0.5601026 | 0.0080127 | 4886.2757 | <.0001* | 0.544398 | 0.5758072 |
| ColorSpace_2[gray] | 0.3012227 | 0.0083462 | 1302.5551 | <.0001* | 0.2848644 | 0.317581 |
| ColorSpace_2[grayTex] | -0.10349 | 0.0072693 | 202.68021 | <.0001* | -0.117737 | -0.089242 |
| ColorSpace_2[lab] | 0.1370965 | 0.0077196 | 315.40177 | <.0001* | 0.1219664 | 0.1522266 |
| ColorSpace_2[labTe | -0.00039 | 0.0077789 | 0.002511 | 0.9600 | -0.015636 | 0.0148566 |
| ColorSpace_2[luv] | 0.1284324 | 0.0077337 | 275.7893 | <.0001* | 0.1132747 | 0.1435901 |
| ColorSpace_2[luvTex] | 0.0013225 | 0.0077721 | 0.0289543 | 0.8649 | -0.013911 | 0.0165556 |
| ColorSpace_2[rgb] | 0.1220584 | 0.0077301 | 249.3261 | <.0001* | 0.1069078 | 0.1372091 |
| nPhases[1] | 0.0382573 | 0.0048623 | 61.907823 | <.0001* | 0.0287274 | 0.0477873 |
| nPhases[2] | 0.0618957 | 0.0045417 | 185.73169 | <.0001* | 0.0529941 | 0.0707972 |
| MaskType[BgDetect] | -0.295998 | 0.0049149 | 3626.9952 | <.0001* | -0.305631 | -0.286365 |
| MaskType[circles] | 0.0146547 | 0.0044621 | 10.786208 | $0.0010^{*}$ | 0.0059091 | 0.0234003 |
| Probability Type[dihedral] | -0.136736 | 0.0067395 | 411.63677 | <.0001* | -0.149945 | -0.123527 |
| Probability Type[EuclidDihedral] | -0.139638 | 0.0065041 | 460.93748 | <.0001* | -0.152386 | -0.126891 |
| Probability Type[mahalanobis] | -0.224777 | 0.0067747 | 1100.822 | <.0001* | -0.238055 | -0.211498 |
| Probability Type[nEuclidDihedral] | -0.146611 | 0.0066815 | 481.48934 | <.0001* | -0.159706 | -0.133515 |
| Probability Type[normEuclidean] | -0.041096 | 0.007604 | 29.209112 | <.0001* | -0.056 | -0.026193 |
| Scale | 0.2454432 | 0.001197 | 42047.99 | <.0001* | 0.2430972 | 0.2477892 |

Figure H.1.5: Statistical analysis general regression of Levelset results ( $F$ measure).


Figure H.1.6: Statistical analysis general regression of Levelset results (F measure).

| Term | Estimate | Std Error | Wald <br> ChiSquare | Prob > <br> ChiSquare | Lower 95\% | Upper 95\% |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| Intercept | 0.7379012 | 0.0077773 | 9002.0163 | <.0001* | 0.722658 | 0.7531444 |
| ColorSpace_2[gray] | 0.1012835 | 0.0078309 | 167.28485 | <.0001* | 0.0859353 | 0.1166318 |
| ColorSpace_2[grayTex] | -0.024995 | 0.007043 | 12.594341 | 0.0004* | -0.038799 | -0.011191 |
| ColorSpace_2[lab] | 0.0380829 | 0.0074363 | 26.226831 | <.0001* | 0.023508 | 0.0526578 |
| ColorSpace_2[labTex] | 0.0009618 | 0.0072313 | 0.0176886 | 0.8942 | -0.013211 | 0.0151348 |
| ColorSpace_2[luv] | 0.0323705 | 0.0074189 | 19.037871 | <.0001* | 0.0178297 | 0.0469113 |
| ColorSpace_2[luvTex] | 0.0005302 | 0.0072295 | 0.0053792 | 0.9415 | -0.013639 | 0.0146998 |
| ColorSpace_2[rgb] | 0.0279392 | 0.0074516 | 14.058185 | 0.0002* | 0.0133343 | 0.0425441 |
| n Phases[1] | 0.0110827 | 0.0046161 | 5.7641316 | 0.0164* | 0.0020352 | 0.0201301 |
| nPhases[2] | 0.0121871 | 0.0046435 | 6.8881628 | 0.0087* | 0.0030859 | 0.0212882 |
| MaskType[BgDetect] | -0.134725 | 0.0045608 | 872.60463 | <.0001* | -0.143664 | -0.125786 |
| MaskType[circles] | 0.0022082 | 0.0046738 | 0.2232097 | 0.6366 | -0.006952 | 0.0113687 |
| ProbabilityType[dihedral] | -0.084381 | 0.0066 | 163.45513 | <.0001* | -0.097317 | -0.071445 |
| ProbabilityType[EuclidDihedral] | -0.075606 | 0.0065896 | 131.64179 | <.0001* | -0.088521 | -0.06269 |
| ProbabilityType[mahalanobis] | -0.115571 | 0.0064072 | 325.35467 | <.0001* | -0.128128 | -0.103013 |
| ProbabilityType[nEuclidDihedral] | -0.084258 | 0.0065704 | 164.44976 | <.0001* | -0.097135 | -0.07138 |
| ProbabilityType[normEuclidean] | -0.045777 | 0.0067678 | 45.750965 | <.0001* | -0.059042 | -0.032512 |
| Scale | 0.2367743 | 0.0016846 | 19754.196 | <.0001* | 0.2334725 | 0.2400762 |

Figure H.1.7: Statistical analysis general regression of Levelset results (PRI).


Figure H.1.8: Statistical analysis general regression of Levelset results (PRI).

|  |  |  | Wald | Prob <br>  <br> Term | Estimate | Std Error | ChiSquare |
| :--- | ---: | ---: | ---: | ---: | ---: | ---: | ---: |
| ChiSquare | Lower 95\% | Upper 95\% |  |  |  |  |  |
| Intercept | 0.4534462 | 0.0033646 | 18163.194 | $<.0001^{*}$ | 0.4468518 | 0.4600407 |  |
| ComponentWeight_1 | 0.4993015 | 0.0018935 | 69530.054 | $<.0001^{*}$ | 0.4955902 | 0.5030128 |  |
| ComponentWeight_2 | 0 | 0 | 0 | 1.0000 | 0 | 0 |  |
| ComponentWeight_3 | -0.496815 | 0.0018954 | 68702.906 | $<.0001^{*}$ | -0.50053 | -0.4931 |  |
| WeightChange_1 | 0 | 0 | 0 | 1.0000 | 0 | 0 |  |
| WeightChange_2 | 0 | 0 | 0 | 1.0000 | 0 | 0 |  |
| WeightChange_3 | 0 | 0 | 0 | 1.0000 | 0 | 0 |  |
| Scale | 0.271892 | 0.0004133 | 432829.65 | $<.0001^{*}$ | 0.271082 | 0.272702 |  |

Figure H.1.9: Statistical analysis general regression of Levelset weight results (F measure).


Figure H.1.10: Statistical analysis general regression of Levelset weight results (F measure).

| Term | Estimate | Std Error | Wald <br> ChiSquare | Prob > <br> ChiSquare | Lower 95\% | Upper 95\% |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| Intercept | 0.6433894 | 0.0026035 | 61070.543 | <.0001* | 0.6382866 | 0.6484922 |
| ComponentWeight_1 | 0.1826486 | 0.0016716 | 11938.343 | <.0001* | 0.1793723 | 0.185925 |
| ComponentWeight_2 | 0 | 0 | 0 | 1.0000 | 0 | 0 |
| ComponentWeight_3 | -0.179015 | 0.0016705 | 11484.412 | <.0001* | -0.182289 | -0.175741 |
| WeightChange_1 | 0 | 0 | 0 | 1.0000 | 0 | 0 |
| WeightChange_2 | 0 | 0 | 0 | 1.0000 | 0 | 0 |
| WeightChange_3 | 0 | 0 | 0 | 1.0000 | 0 | 0 |
| Scale | 0.2400987 | 0.0006014 | 159370.14 | <.0001* | 0.2389199 | 0.2412775 |

Figure H.1.11: Statistical analysis general regression of Levelset weight results (PRI).


Figure H.1.12: Statistical analysis general regression of Levelset weight results (PRI).

|  | Wald |  |  |  |  |  | Prob > |
| :--- | ---: | ---: | ---: | ---: | ---: | ---: | ---: |
| Term | Estimate | Std Error | ChiSquare | ChiSquare | Lower 95\% | Upper 95\% |  |
| Intercept | 0.7512258 | 0.0023404 | 103027.53 | $<.0001^{*}$ | 0.7466386 | 0.7558129 |  |
| nPhases[1] | 0.0167857 | 0.0005809 | 834.87826 | $<.0001^{*}$ | 0.0156471 | 0.0179243 |  |
| MaskType[BgDetect] | 0.029611 | 0.0005809 | 2598.0605 | $<.0001^{*}$ | 0.0284723 | 0.0307496 |  |
| MaskRadius | 0.0005089 | $9.1769 e-5$ | 30.752159 | $<.0001^{*}$ | 0.000329 | 0.0006888 |  |
| MaskPadding | -0.000391 | $9.1745 e-5$ | 18.149918 | $<.0001^{*}$ | -0.000571 | -0.000211 |  |
| MaskShift | $9.2574 \mathrm{e}-5$ | $9.1761 e-5$ | 1.0178034 | 0.3130 | $-8.727 \mathrm{e}-5$ | 0.0002724 |  |
| Scale | 0.1289192 | 0.0002027 | 404649.22 | $<.0001^{*}$ | 0.128522 | 0.1293164 |  |

Figure H.1.13: Statistical analysis general regression of Levelset init results (F measure).


Figure H.1.14: Statistical analysis general regression of Levelset init results (F measure).

|  | Wald |  |  |  |  |  | Prob > |
| :--- | ---: | ---: | ---: | ---: | ---: | ---: | ---: |
| Term | Estimate | Std Error | ChiSquare | ChiSquare | Lower 95\% | Upper 95\% |  |
| Intercept | 0.7655588 | 0.0041679 | 33738.278 | $<.0001^{*}$ | 0.7573899 | 0.7737277 |  |
| nPhases[1] | -0.00718 | 0.001037 | 47.949414 | $<.0001^{*}$ | -0.009213 | -0.005148 |  |
| MaskType[BgDetect] | 0.0256131 | 0.001037 | 610.1061 | $<.0001^{*}$ | 0.0235807 | 0.0276455 |  |
| MaskRadius | -0.00026 | 0.000164 | 2.5225437 | 0.1122 | -0.000582 | $6.0951 e-5$ |  |
| MaskPadding | 0.0002653 | 0.0001639 | 2.6198794 | 0.1055 | -0.000056 | 0.0005866 |  |
| MaskShift | -0.000115 | 0.000164 | 0.4902262 | 0.4838 | -0.000436 | 0.0002066 |  |
| Scale | 0.2301175 | 0.0006471 | 126450.29 | $<.0001^{*}$ | 0.2288492 | 0.2313859 |  |

Figure H.1.15: Statistical analysis general regression of Levelset init results (PRI).


Figure H.1.16: Statistical analysis general regression of Levelset init results (PRI).

| Model | AICc | AICc Weight | . 2.4 | . 6.8 | BIC | SSE | MSE | RMSE | R-Square |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| Quintic | -360.627 | 0.9567884 |  |  | -332.846 | 9.4857144 | 0.0236551 | 0.1538023 | 0.6822377 |
| Quartic | -354.4321 | 0.0432116 |  |  | -330.5892 | 9.6803265 | 0.0240804 | 0.1551787 | 0.6757184 |

Figure H.1.17: Statistical analysis general regression of gPb -owt results ( F measure).


Figure H.1.18: Statistical analysis general regression of gPb-owt results ( F measure).

| Model | AICc | AICc Weight | . 2.4 . 6.8 | BIC | SSE | MSE | RMSE | R-Square |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| Quartic | -66.52588 | 0.7374108 |  | -42.683 | 19.638303 | 0.0488515 | 0.2210238 | 0.2343157 |
| Quintic | -64.46077 | 0.2625892 | ! | -36.67978 | 19.638033 | 0.0489727 | 0.2212977 | 0.2343262 |

Figure H.1.19: Statistical analysis general regression of gPb -owt results (PRI).


Figure H.1.20: Statistical analysis general regression of gPb-owt results (PRI).

## H.1.1.3 Best parameter sets

Based on the plotted results from each scoring metric and regression the following parameters were determined to be the best.

- Absolute - From PR plot (min D2P, max F, max PRI, min LD)
- Relative - From statistical regression analysis (max PRI with max F as tie breaker)
- kMean Base
- Absolute (D2P) - Lab, Squared Euclidean, Uniform, 2 Clusters
- Absolute (F) - Gray, Squared Euclidean, Uniform, 2 Clusters
- Absolute (PRI) - Luv, Squared Euclidean, Uniform, 4 Clusters
- Absolute (LD) - Luv, Squared Euclidean, Uniform, 2 Clusters
- Relative (Reg-PRI) - Luv, Standardized Euclidean, Cluster, 3 clusters
- LevelSet Base
- Absolute (D2P) - Lab, Squared Euclidean, Background Detect, 1 Phase
- Absolute (F) - RGB, Euclidean + Dihedral Angle, kMean Clustering, 1 Phase
- Absolute (PRI) - Luv, Squared Euclidean, Circles, 3 Phases
- Absolute (LD) - Luv, Squared Euclidean, Circles, 1 Phase
- Relative (Reg-PRI) - Gray, Squared Euclidean, Circles, 2 Phase
- LevelSet Weight
- Absolute (D2P) - [1.5, 1.5, 1] Weight, [0.8, 0.8, 0.8] Weight Change
- Absolute (F) - [1.5, 1.5, 1] Weight, [0.8, 0.8, 0.8] Weight Change
- Absolute (PRI) - [1.5, 1, 1] Weight, [0.8, 0.8, 0.8] Weight Change
- Absolute (LD) - [1.5, 1.5, 1] Weight, [0.8, 0.8, 0.8] Weight Change
- Relative (Reg-PRI) - [1.5, x, 0.5] Weight, [x, x, x] Weight Change
- LevelSet Init
- Absolute (D2P) - Background Detect, 5p Radius, 10p Padding, 12p Shift, 1 Phase
- Absolute (F) - Background Detect, 5p Radius, 10p Padding, 12p Shift, 1 Phase
- Absolute (PRI) - Circles, 5p Radius, 17p Padding, 13p Shift, 2 Phases
- Absolute (LD) - Circles, 10p Radius, 19p Padding, 12p Shift, 1 Phase
- Relative (Reg-PRI) - Background Detect, 10p Radius, 15p Padding, 17p Shift, 1 Phase
- LevelSet Noise
- Absolute (D2P) - Spatial scale 1, monogenic signal gradient, gradient size 3, gradient smooth sigma 0.2 , gradient weight [1.5,0.5,0.5]
- Absolute (F) - Spatial scale 1, monogenic signal gradient, gradient size 6, gradient smooth sigma 0.6 , gradient weight $[1.5,0.5,0.5]$
- Absolute (PRI) - Spatial scale 13, monogenic signal gradient, gradient size 6, gradient smooth sigma 0.4 , gradient weight [0.5,1,1.5]
- Absolute (LD) - Spatial scale 1, monogenic signal gradient, gradient size 3, gradient smooth sigma 0.4 , gradient weight [1.5, $1,0.5$ ]
- Relative (Reg-PRI) - Spatial scale 4, gPb gradient, gradient size 6
- LevelSet MS
- Absolute (D2P) - 1 phase, no cleanup, overlapping merge, 2k k-mean split, initial phase select
- Absolute (F) - 2 phases, no cleanup, no merge, force split, 2k k-mean split, iterative phase select
- Absolute (PRI) - 2 phases, background cleanup, no merge, iterative k-mean split, iterative phase select
- Absolute (LD) - 1 phase, no cleanup, overlapping merge, no split, initial phase select
- Relative (Reg-PRI) - 2 Phases, no cleanup, overlapping merge, distance merge, force split, histogram split, iterative phase select
- gPb-owt-ucm
- Absolute Global (D2P) - 0.4 Threshold
- Absolute Global (F) - 0.2 Threshold
- Absolute Global (PRI) - 0.3 Threshold
- Absolute Global (LD) - 0.2 Threshold
- Relative (Reg-PRI) - 0.2 Threshold


## H.1.1.4 Example Segmentation

Images generated from the parameter bests: Distance to Pro (D2P), F-Measure (F), Probabilistic Rand Index (PRI), Loomis Distance (LD), Regression Model (RG). The Loomis filtered images are a visual representation of how the images would feel haptically.


(k) LevelSet Base

(I) kMean Base (Loomis Filter)

(m) kMean Base

(n) LevelSet Weight (Loomis Filter)

(o) LevelSet Init

D2P

(p) LevelSet Init (Loomis Filter)

Figure H.1.21: Example segmentation boundaries using the best of kMean, LevelSet, and gPb-owtucm compared to human segmentation


(k) LevelSet Base

(I) kMean Base (Loomis Filter)

(m) kMean Base

(n) LevelSet Weight (Loomis Filter)

(o) LevelSet Init

(p) LevelSet Init (Loomis Filter)

Figure H.1.22: Example segmentation boundaries using the best of kMean, LevelSet, and gPb-owtucm compared to human segmentation


-
(b) Orig
(c) \#1
(d) \#1
(e) \#2 (f) \#2 -

D2P F
(g) gPb-owt-ucm

| $\circ$ | $\circ$ |
| :--- | :---: |
| 08 | $0 \nabla$ |
| D2P | F |


| $\circ$ | 0 | 0 |
| :--- | :--- | :--- |
| 08 | $\theta \nabla$ |  |
| PRI | LD | RG | (h) gPb-owt-ucm (Loomis Filter)



D2P


F

8
LD
(i) kMean Base


LD


RG
(j) kMean Base (Loomis Filter)


Figure H.1.23: Example segmentation boundaries using the best of kMean, LevelSet, and gPb-owtucm compared to human segmentation
(a)

(a) Orig
(b) Orig
(c) \#1
(d) \#1
(e) \#2
(f) \#2

(g) gPb-owt-ucm

(h) gPb-owt-ucm (Loomis Filter)
D2P

(i) kMean Base


(k) LevelSet Base

(I) kMean Base (Loomis Filter)

D2P

F

PRI

(m) kMean Base
 D2P

F


PRI




RG
(n) LevelSet Weight (Loomis Filter)





(o) LevelSet Init


Figure H.1.24: Example segmentation boundaries using the best of kMean, LevelSet, and gPb-owtucm compared to human segmentation

## H. 2 Simplification

## H.2.1 Regression Selection

## H.2.1.1 Cue List

Local cues - each pixel has a value

- X position of pixel from center
- Y position of pixel from center
- R position of pixel from center
- $\Theta$ position of pixel from center
- Red color depth of pixel
- Green color depth of pixel
- Blue color depth of pixel
- Brightness depth of pixel
- Red-Green color depth of pixel
- Blue-Yellow color depth of pixel
- Sum texture local entropy of Luv depths
- Sum texture third moment of Luv depths
- Monogenic signal energy of red
- Monogenic signal energy of green
- Monogenic signal energy of blue depth
- Max monogenic signal energy of RGB
- Local entropy of max monogenic signal energy of RGB
- Monogenic signal energy of brightness
- Monogenic signal energy of red-green
- Monogenic signal energy of blue-yellow
- Monogenic signal energy of texture local entropy of Luv
- Monogenic signal energy of texture third moment of Luv
- Max monogenic signal of Luv+tm energy
- Local entropy of max monogenic signal energy of Luv+tm
- SUSAN energy filter of red color
- SUSAN energy filter of green color
- SUSAN energy filter of blue color
- Max SUSAN energy filter of RGB color
- SUSAN energy filter of brightness
- SUSAN energy filter of red-green
- SUSAN energy filter of blue-yellow
- Max SUSAN energy filter of Luv+tm
- Multi scale gradient of brightness size 1
- Multi scale gradient of brightness size 2
- Multi scale gradient of brightness size 3
- Multi scale gradient of red-green size 1
- Multi scale gradient of red-green size 2
- Multi scale gradient of red-green size 3
- Multi scale gradient of blue-yellow size 1
- Multi scale gradient of blue-yellow size 2
- Multi scale gradient of blue-yellow size 3
- Multi scale gradient of textons size 1
- Multi scale gradient of textons size 2
- Multi scale gradient of textons size 3
- Multi scale gradient probability of boundary
- Spectral probability of boundary
- Global probability of boundary
- Local entropy of red
- Local entropy of green
- Local entropy of blue
- Sum of local entropy of RGB
- Local entropy of brightness
- Local entropy of red-green
- Local entropy of blue-yellow
- Sum of local entropy of Luv+tm
- SUSAN texture of red
- SUSAN texture of green
- SUSAN texture of blue
- Sum of SUSAN texture of RGB
- SUSAN texture of brightness
- SUSAN texture of red-green
- SUSAN texture of blue-yellow
- Sum of SUSAN texture of Luv+tm
- Orientation of max monogenic signal energy of RGB
- Orientation of max monogenic signal energy of Luv+tm
- Orientation of multi scale gradient of brightness size 1
- Orientation of multi scale gradient of brightness size 2
- Orientation of multi scale gradient of brightness size 3
- Orientation of multi scale gradient of redgreen size 1
- Orientation of multi scale gradient of redgreen size 2
- Orientation of multi scale gradient of redgreen size 3
- Orientation of multi scale gradient of blue-yellow size 1
- Orientation of multi scale gradient of blue-yellow size 2
- Orientation of multi scale gradient of blue-yellow size 3
- Orientation of multi scale gradient of textons size 1
- Orientation of multi scale gradient of textons size 2
- Orientation of multi scale gradient of tex-
tons size 3
- Phase of max monogenic signal energy of RGB
- Phase of max monogenic signal energy of Luv+tm
- Canny edge detection
- Shen edge detection
- SUSAN edge detection
- Min Eigen vector corner detection
- Harris corner detection
- SUSAN corner detection
- Global probability of boundary
- Global probability of boundary (gPb-owtucm)
- Levelset boundary
- Distance from pixel to Canny edge
- Distance from pixel to Shen edge
- Distance from pixel to SUSAN edge
- Distance from pixel to min Eigen corner
- Distance from pixel to Harris corner
- Distance from pixel to SUSAN corner
- Distance from pixel to global probability of boundary
- Distance from pixel to levelset boundary
- Curvature of Canny edge
- Curvature of Shen edge
- Curvature of SUSAN edge
- Curvature of global probability of boundary
- Curvature of levelset boundary
- Probability of edge vs not edge (2 stage)
- Logical likely edge vs not edge (2 stage)

Region cues - each pixel in the same region share the same value

- Percent of image
- Percent of segment
- Eccentricity
- Euler Number
- Solidity
- Perimeter
- Area perimeter ratio
- Center mass X position
- Center mass Y position
- Center mass R position
- Center mass $\Theta$ position
- Center mass Second moment
- Center mass Orientation angle
- Standard deviation of $X$ position
- Standard deviation of Y position
- Standard deviation of R position
- Standard deviation of $\Theta$ position
- Mean red
- Mean green
- Mean blue
- Mean brightness
- Mean red-green
- Mean blue-yellow
- Mean local entropy of Luv
- Mean third moment
- Standard deviation of red
- Standard deviation of green
- Standard deviation of blue
- Standard deviation of brightness
- Standard deviation of red-green
- Standard deviation of blue-yellow
- Standard deviation of local entropy of Luv
- Standard deviation of third moment
- Mean local entropy red
- Mean local entropy green
- Mean local entropy blue
- Mean local entropy brightness
- Mean local entropy red-green
- Mean local entropy blue-yellow
- Mean local entropy monogenic energy red
- Mean local entropy monogenic energy green
- Mean local entropy monogenic energy blue
- Mean local entropy monogenic energy brightness
- Mean local entropy monogenic energy red-green
- Mean local entropy monogenic energy
blue-yellow
- Logical in/out landmark box
- Logical in/out face detection
- Logical in/out face detection tight

Segment cues - each pixel in the same segment share the same value

- Percent of image
- Eccentricity
- Euler Number
- Solidity
- Perimeter
- Area perimeter ratio
- Mean red
- Mean green
- Mean blue
- Mean brightness
- Mean red-green
- Mean blue-yellow
- Mean local entropy of Luv
- Mean third moment
- Standard deviation of red
- Standard deviation of green
- Standard deviation of blue
- Standard deviation of brightness
- Standard deviation of red-green
- Standard deviation of blue-yellow
- Standard deviation of local entropy of Luv
- Standard deviation of third moment
- Center mass X position
- Center mass Y position
- Center mass R position
- Center mass $\Theta$ position
- Center mass Second moment
- Center mass Orientation angle
- Number of regions in segment
- Mean region area
- Standard deviation region area
- Mean local entropy red
- Mean local entropy green
- Mean local entropy blue
- Mean local entropy brightness
- Mean local entropy red-green
- Mean local entropy blue-yellow
- Mean local entropy monogenic energy red
- Mean local entropy monogenic energy green
- Mean local entropy monogenic energy blue
- Mean local entropy monogenic energy brightness
- Mean local entropy monogenic energy red-green
- Mean local entropy monogenic energy blue-yellow

Global cues - each pixel in the same image share the same value

- Mean red
- Mean green
- Mean blue
- Mean brightness
- Mean red-green
- Mean blue-yellow
- Mean local entropy of Luv
- Mean third moment
- Standard deviation of red
- Standard deviation of green
- Standard deviation of blue
- Standard deviation of brightness
- Standard deviation of red-green
- Standard deviation of blue-yellow
- Standard deviation of local entropy of Luv
- Standard deviation of third moment
- Mean local entropy red
- Mean local entropy green
- Mean local entropy blue
- Mean local entropy brightness
- Mean local entropy red-green
- Mean local entropy blue-yellow
- Number of regions in segment
- Mean region area
- Standard deviation region area
- Number of segments in segment
- Mean segment area
- Standard deviation segment area
- Logical landmark present
- Logical face present
- Logical image is a cartoon

Target cues - Calculated from professional diagrams acting as a model target

- Logical edge
- Probability of segment accuracy
- Probability of edge
- Probability of texture
- Probability of region accuracy


## H.2.2 Edge Anticipation



Figure H.2.1: LASSO Coefficients


Figure H.2.2: LASSO MSE


Figure H.2.3: GLM Leverage


Figure H.2.4: GLM Cook's Distance


Figure H.2.5: GLM Residuals


Figure H.2.6: GLM Residuals Probability


Figure H.2.7: GLM Fit

Table H.2.1: Simplification edge modeling variable significant.

| Variable | Estimate | SE | tStat | pValue |
| :--- | ---: | ---: | ---: | ---: |
| (Intercept) | 0.04 | 0.00017 | 232.39 | 0 |
| Local R Position | -0.10 | 0.00024 | -386.67 | 0 |
| Local entropy of monogenic signal energy | -0.06 | 0.00066 | -91.24 | 0 |
| Local multi scale gradient brightness size 2 | -0.04 | 0.00134 | -28.03 | $\sim 0$ |
| Local multi scale gradient blue-yellow size 1 | 0.32 | 0.00114 | 278.32 | 0 |
| Local multi scale gradient texture size 1 | -0.08 | 0.00077 | -107.71 | 0 |
| Local multi scale gradient texture size 2 | 0.11 | 0.00086 | 124.42 | 0 |
| Local SUSAN texture red | 0.05 | 0.00043 | 106.19 | 0 |
| Local SUSAN texture luv | 0.09 | 0.00062 | 151.40 | 0 |
| Local Min Eigen vector corner | 0.06 | 0.00050 | 118.41 | 0 |
| Local Harris corner detection | 0.03 | 0.00052 | 50.34 | 0 |
| gPb boundary | 0.25 | 0.00032 | 796.11 | 0 |
| Level set boundary | 0.09 | 0.00028 | 336.32 | 0 |
| Global mean region area | 0.01 | 0.00054 | 6.72 | $\sim 0$ |
| Local entropy of monogenic signal energy ${ }^{2}$ | 0.25 | 0.00087 | 287.71 | 0 |
| Local multi scale gradient blue-yellow size 1 ${ }^{2}$ | -0.39 | 0.00176 | -223.65 | 0 |
| Local multi scale gradient texture size 1 ${ }^{2}$ | 0.11 | 0.00093 | 121.14 | 0 |
| Local multi scale gradient texture size 2 ${ }^{2}$ | -0.08 | 0.00106 | -78.36 | 0 |

- $\operatorname{Adj}-R^{2}-0.7625$
- Initial outlier pixels - 0.14\%
- Edge outliers - 0.05\%
- Outlier Images - 7.90\%
- Total outliers - 8.63\%


## H.2.3 Region Suppression



Figure H.2.8: LASSO Coefficients


Figure H.2.9: LASSO MSE


Figure H.2.10: GLM Leverage


Figure H.2.11: GLM Cook's Distance


Figure H.2.12: GLM Residuals


Figure H.2.13: GLM Residuals Probability


Figure H.2.14: GLM Fit

Table H.2.2: Simplification region modeling variable significant. Part 1

| Variable | Estimate | SE | tStat | pValue |
| :--- | ---: | ---: | ---: | ---: |
| (Intercept) | 0.75 | 0.00112 | 664.96 | 0 |
| Local R Position | -0.02 | 0.00036 | -49.36 | 0 |
| Local brightness monogenic signal energy | -0.04 | 0.00418 | -9.82 | 0 |
| Local luv monogenic signal energy | 0.07 | 0.00368 | 18.73 | 0 |
| Level set boundary | -0.04 | 0.00031 | -122.13 | 0 |
| Region percent of image | 0.03 | 0.00054 | 52.70 | 0 |
| Region percent of segment | 0.49 | 0.00097 | 504.12 | 0 |
| Region Eccentricity | -0.10 | 0.00032 | -315.95 | 0 |
| Region R position | -1.05 | 0.00241 | -437.43 | 0 |
| Region R position stdv | -0.02 | 0.00251 | -6.67 | 0 |
| Region 3rd moment | -0.44 | 0.00073 | -610.21 | 0 |
| Region Red stdv | -0.01 | 0.00038 | -20.16 | 0 |
| Region red energy | 0.39 | 0.00145 | 270.55 | 0 |
| Region brightness monogenic energy | 0.72 | 0.00490 | 147.41 | 0 |
| Segment r position | 0.14 | 0.00231 | 61.19 | 0 |
| Segment number of contained regions | -0.09 | 0.00039 | -225.60 | 0 |
| Segment stdv of contained region areas | 0.02 | 0.00041 | 58.40 | 0 |
| Segment brightness monogenic energy | -0.35 | 0.00274 | -129.53 | 0 |
| Local brightness monogenic signal energy ${ }^{2}$ | -0.07 | 0.00504 | -14.78 | 0 |
| Local luv monogenic signal energy ${ }^{2}$ | -0.14 | 0.00229 | 59.54 | 0 |
| Region mean red entropy ${ }^{2}$ | -0.27 | 0.00150 | -179.50 | 0 |
| Region brightness monogenic energy ${ }^{2}$ | -1.04 | 0.00498 | -208.85 | 0 |
| Segment brightness monogenic energy ${ }^{2}$ | -0.02 | 0.00326 | -6.53 | 0 |

Table H.2.3: Simplification region modeling variable significant. Part 2

| Variable | Estimate | SE | tStat | pValue |
| :---: | :---: | :---: | :---: | :---: |
| Local brightness monogenic signal energy |  |  |  |  |
| Region R position | 1.06 | 0.01098 | 96.74 | 0 |
| Local luv monogenic signal energy : |  |  |  |  |
| Region R position | -0.31 | 0.01258 | -24.68 | 0 |
| Local luv monogenic signal energy : |  |  |  |  |
| Region R position stdv | -0.47 | 0.01234 | -37.85 | 0 |
| Local luv monogenic signal energy |  |  |  |  |
| Region R position stdv | 0.15 | 0.00331 | 44.76 | 0 |
| Local brightness monogenic signal energy |  |  |  |  |
| Region brightness energy | -0.07 | 0.00359 | -19.29 | 0 |
| Local brightness monogenic signal energy |  |  |  |  |
| Region brightness monogenic energy | 0.19 | 0.00917 | 21.13 | 0 |
| Local luv monogenic signal energy : |  |  |  |  |
| Region brightness monogenic energy | 0.46 | 0.00792 | 58.42 | 0 |
| Local brightness monogenic signal energy |  |  |  |  |
| Segment r position | -0.36 | 0.00695 | -52.43 | 0 |
| Local luv monogenic signal energy |  |  |  |  |
| Segment r position | -0.28 | 0.00733 | -38.66 | 0 |
| Region R position : |  |  |  |  |
| Segment r position | 1.11 | 0.00207 | 536.58 | 0 |
| Region R position stdv : |  |  |  |  |
| Segment r position | -1.06 | 0.00286 | -370.27 | 0 |
| Region red energy |  |  |  |  |
| Segment r position | -1.15 | 0.00147 | -780.79 | 0 |
| Region brightness monogenic energy |  |  |  |  |
| Segment r position | 1.04 | 0.00352 | 296.33 | 0 |
| Local brightness monogenic energy |  |  |  |  |
| Segment brightness monogenic energy | -0.42 | 0.00652 | -63.90 | 0 |
| Local brightness monogenic energy |  |  |  |  |
| Segment brightness monogenic energy | -0.36 | 0.00643 | -56.11 | 0 |
| Region red energy : |  |  |  |  |
| Segment brightness monogenic energy | 0.50 | 0.00446 | 112.16 | 0 |
| Region R position stdv : |  |  |  |  |
| Segment brightness monogenic energy | 0.76 | 0.00429 | 178.30 | 0 |
| Region red energy : |  |  |  |  |
| Segment brightness monogenic energy | -0.24 | 0.00157 | -149.28 | 0 |
| Region brightness monogenic energy : |  |  |  |  |
| Segment brightness monogenic energy | 0.60 | 0.00543 | 109.73 | 0 |

- $\operatorname{Adj}-R^{2}-0.8459$
- Initial outlier pixels - 0.82\%
- Outlier Images - 7.90\%
- Total outliers - 9.99\%


## H.2.4 Segment Texturing



Figure H.2.15: LASSO Coefficients


Figure H.2.16: LASSO MSE


Figure H.2.17: GLM Leverage


Figure H.2.18: GLM Cook's Distance


Figure H.2.19: GLM Residuals


Figure H.2.20: GLM Residuals Probability


Figure H.2.21: GLM Fit

Table H.2.4: Simplification texture modeling variable significant. Part 1

| Variable | Estimate | SE | tStat | pValue |
| :--- | ---: | ---: | ---: | ---: |
| (Intercept) | 0.80 | 0.00170 | 471.60 | 0 |
| Local R position | -0.10 | 0.00049 | -209.07 | 0 |
| Local color blue-yellow | -0.14 | 0.00032 | -439.05 | 0 |
| Local brightness monogenic energy | 1.30 | 0.00746 | 174.70 | 0 |
| Local luv mongenic energy | -0.65 | 0.00599 | -108.64 | 0 |
| Local green entropy | -1.34 | 0.00627 | -214.28 | 0 |
| Local luv entropy | 1.43 | 0.00726 | 196.52 | 0 |
| Local luv monogenic orientation | 0.02 | 0.00024 | 72.42 | 0 |
| Region percent of image | -0.36 | 0.00099 | -362.24 | 0 |
| Region Euler Number | 0.79 | 0.00072 | 1094.30 | 0 |
| Region R position | -0.75 | 0.00445 | -168.83 | 0 |
| Region R position stdv | 0.14 | 0.00422 | 33.63 | 0 |
| Region brightness monogenic energy | 0.02 | 0.00804 | 3.034 | 0 |
| Segment percent of image | -0.12 | 0.00096 | -126.85 | 0 |
| Segment R position | -0.04 | 0.00367 | -10.55 | 0 |
| Segment brightness monogenic energy | 0.11 | 0.00429 | 25.94 | 0 |
| Global luv entropy | -0.06 | 0.00067 | -92.37 | 0 |
| Global red energy | -0.46 | 0.00051 | -895.31 | 0 |
| Local brightness monogenic energy ${ }^{2}$ | -0.37 | 0.00812 | -45.75 | 0 |
| Local luv mongenic energy ${ }^{2}$ | 0.49 | 0.00300 | 165.29 | 0 |
| Region brightness monogenic energy ${ }^{2}$ | -0.35 | 0.00951 | -36.51 | 0 |
| Segment brightness monogenic energy ${ }^{2}$ | 2.41 | 0.00530 | 454.71 | 0 |

Table H.2.5: Simplification texture modeling variable significant. Part 2

| Variable | Estimate | SE | tStat | pValue |
| :---: | :---: | :---: | :---: | :---: |
| Local brightness monogenic energy : |  |  |  |  |
| Region R position | -3.74 | 0.02248 | -166.40 | 0 |
| Local luv mongenic energy : |  |  |  |  |
| Region R position | 4.10 | 0.02333 | 175.57 | 0 |
| Local green entropy |  |  |  |  |
| Region R position | 4.29 | 0.02015 | 212.95 | 0 |
| Local luv entropy : |  |  |  |  |
| Region R position | -0.54 | 0.02213 | -24.27 | 0 |
| Local brightness monogenic energy : |  |  |  |  |
| Region R position stdv | 3.52 | 0.02137 | 164.89 | 0 |
| Local luv mongenic energy : |  |  |  |  |
| Region R position stdv | -3.67 | 0.02238 | -163.97 | 0 |
| Local green entropy : |  |  |  |  |
| Region R position stdv | -1.92 | 0.01777 | -107.83 | 0 |
| Local luv entropy : |  |  |  |  |
| Region R position stdv | -1.34 | 0.01934 | -69.47 | 0 |
| Local brightness monogenic energy : |  |  |  |  |
| Region brightness monogenic energy | -2.89 | 0.01791 | -161.19 | 0 |
| Local luv mongenic energy : |  |  |  |  |
| Region brightness monogenic energy | 1.57 | 0.01313 | 119.50 | 0 |
| Local green entropy : |  |  |  |  |
| Region brightness monogenic energy | 4.25 | 0.01846 | 230.19 | 0 |
| Local luv entropy |  |  |  |  |
| Region brightness monogenic energy | -3.18 | 0.02127 | -149.34 | 0 |
| Local luv mongenic energy : |  |  |  |  |
| Segment R position | -1.57 | 0.01340 | -117.35 | 0 |
| Local green entropy |  |  |  |  |
| Segment R position | -1.88 | 0.01230 | -152.65 | 0 |
| Local luv entropy : |  |  |  |  |
| Segment R position | 1.19 | 0.01395 | 84.45 | 0 |

Table H.2.6: Simplification texture modeling variable significant. Part 3

| Variable | Estimate | SE | tStat | pValue |
| :---: | :---: | :---: | :---: | :---: |
| Region R position : |  |  |  |  |
| Segment R position | -0.48 | 0.00357 | -134.28 | 0 |
| Region R position stdv : |  |  |  |  |
| Segment R position | -0.03 | 0.00427 | -5.91 | 0 |
| Region brightness monogenic energy : |  |  |  |  |
| Segment R position | 1.37 | 0.00620 | 221.42 | 0 |
| Local brightness monogenic energy : |  |  |  |  |
| Segment brightness monogenic energy | 1.15 | 0.01344 | 85.21 | 0 |
| Local luv mongenic energy : |  |  |  |  |
| Segment brightness monogenic energy | -1.36 | 0.01094 | -124.20 | 0 |
| Local green entropy |  |  |  |  |
| Segment brightness monogenic energy | -3.68 | 0.01419 | -259.52 | 0 |
| Local luv entropy : |  |  |  |  |
| Segment brightness monogenic energy | 1.92 | 0.01635 | 117.19 | 0 |
| Region R position : |  |  |  |  |
| Segment brightness monogenic energy | -2.21 | 0.00757 | -291.56 | 0 |
| Region R position stdv : |  |  |  |  |
| Segment brightness monogenic energy | 2.94 | 0.00709 | 414.18 | 0 |
| Region brightness monogenic energy : |  |  |  |  |
| Segment brightness monogenic energy | -1.84 | 0.01068 | -172.61 | 0 |
| Region brightness monogenic energy : |  |  |  |  |
| Global texture | -0.40 | 0.00558 | -71.08 | 0 |
| Local green entropy |  |  |  |  |
| Global texture | 1.54 | 0.00498 | 308.68 | 0 |
| Local luv entropy : |  |  |  |  |
| Global texture | -1.19 | 0.00562 | -211.56 | 0 |

1. $A d j-R^{2}-0.7023$
2. Initial outlier pixels-14.25\%
3. Outlier Images - 0\%
4. Total outliers - 10.80\%

## Appendix I: Proposal

## I. 1 Introduction

Problem and Significance About 285 million people worldwide are blind or visually impaired (BVI) [2]. These individuals often rely heavily on alternatives to visual information for their daily lives. While the written word is relatively easy to convert and print to braille or provide as speech output using computers, graphical information is much more difficult to supply to user's who are BVI. This is because graphical information cannot be simply converted to tactile, or auditory, form in a one-to-one correspondence. Even simple line diagrams require the intervention of a sighted professional to modify the diagram to a simplified and accessible form. Although visual information such as graphs, pictures, diagrams, etc. could simply be described in words, there are several problems with this approach. First, only about $9 \%$ of people who are BVI are able to read braille [3]. Second, if the objects are unfamiliar it may be difficult to understand the description. Third, it reduces the ability to discover spatial patterns or relationships, which are lost when the graphic is turned into words. Fourth, the individuals who create the descriptions may not always interpret the information appropriately. Finally, it can prevent the user from learning as much because they are not forced to make their own inferences. This is especially important for school children as they need to be able to comprehend and retain class material.

Currently, for an individual who is BVI to access a tactile diagram, they must rely on a sighted, professional diagram maker to convert the visual diagram to tactile diagrams. While many schools have access to a teacher for people who are blind or visually impaired (TVI) who can assist with converting visual information to tactile diagrams and written descriptions, this service is limited because of the time it takes to make the graphics. Particularly for courses that heavily use diagrams, not all the diagrams will be pro-
vided in tactile form, limiting learning opportunities. Most adults do not have access to such costly support in their daily lives at work or at home often hindering their abilities to complete assigned tasks, limiting their independence and upward mobility in their job. The severe lack of visual information accessible by people who are BVI may also explain why fewer students who are BVI are attracted to science, technology, engineering, and math (STEM) fields, as these fields heavily involve graphical representations.

For several years, researchers (Ladner et al. \& Barner et al.) have looked at the process of converting visual images to tactile graphics and attempted to devise an automatic conversion algorithm [6-11,20-22]. These research attempts were often limited to specific image types such as vector images and graphs or the user testing and comparison to professionally made diagrams was not fully explored. In general, algorithms with good performance have been developed for optical character recognition of diagram labels and their translation to Braille, as well as representing basic 2-D graphs. There has been less success with more complex types of diagrams as shown below in the output for the commercially available FireBird Graphics editor (the best available), which provides an automatic visual-tactile diagram conversion option.

The focus of this thesis is on automatically translating visual graphics, which may be in drawing or photographic form and depict objects or object scenes, to effective and easily interpreted tactile diagrams. Successful comprehension will be based on the ability of users to interpret the form of an object and/or identify it, and the spatial relationships between objects in an object scene. For these drawings or photographs, we will assume that all objects in the scene are presented as whole objects. We will also assume, given previous results, that labels are successfully removed from the diagrams. We will also not consider the use of context (such as labels, text descriptions, etc.) at this point as it would expand the scope of this project greatly beyond a single thesis given the current state of the art.

## I.1.1 Background

A wide variety of methods/materials can be used to create physical tactile graphics, including: drawing boards, buildup displays, embossed paper, vacuum forming, microcapsule paper, silk screening, etc [4]. Regardless of method, the conversion process is time consuming and often costly. The translating professional generally requires training to ensure they follow specific design principals such as: accounting for proportion/scale, perspective, inclusion/exclusion of material in the diagram, simplification of shapes and concepts, etc. [4]. The choices made during the design are often guided by specific information including: what are the necessary components, who uses the information conveyed by the diagram, how will the diagram be used, what materials should be used, what materials are available, etc [4].

Core Processes of Diagram Translation The primary steps in image conversion are to segment the image into continuous regions (i.e., object parts) and then simplify the image by removing extraneous information and viewing perspective. The initial segmentation must attempt to find the most important parts of the image that need to be conveyed to the user in the same manner the human visual system would focus on important aspects of the image. Image segmentation is distinctly different from simply detecting regions or boundaries in an image. Regions are defined as contiguous close bounded areas of an image while segments may be non-contiguous groups of regions that represent the same part of an image. For example the image of a dog may have 4 regions: background, head, collar, and body; while only having 3 segments: background, dog, and collar; because the dog's head and body together represent the dog as a whole.

Next, the system must attempt to locate and remove any extraneous information. Such information could simply be noise or details that have carried through. However, it can even be other large objects that add too much to the diagram to be understood. Due to the orders of magnitude lower information bandwidth that the tactile system has available
compared to the visual system, any extra information poses a comprehension threat to the user interpreting the data; this bandwidth discrepancy can be seen in Table C.1. At this point it is not clear in which phase of the automatic conversion process perspective should be removed because there is no current research on the topic but it is clear that removing perspective will ease comprehension of the primary information about object shape and object relationships.

## I.1.2 Manual

Many books, web tutorials, and teaching courses exist to help train teachers for individuals who are visually impaired (TVIs) how to develop tactile graphics [4,5,14-18]. These sources teach methods ranging in their development methods from the use of craft supplies, to image manipulation software; some of the hand- made graphics can be viewed in Figure l.1.1. More teachers are looking towards technology in order to produce consistent results and to reduce the time it takes to make tactile graphics. Figure I.1.1 also shows textures that were determined to be salient and distinguishable for use in tactical graphics and line drawings. The use of texture is important in helping the tactile reader mentally separate different regions or group like regions together that would otherwise have no distinguishable relationship.


Figure I.1.1: Example hand-made tactile graphics (from various sources including EmpowerLingua and AFB [4]). Example computer aided tactile graphics textures (not to scale) [18]

## I.1.3 Automatic

Current commercially available software attempts to allow users to develop their own tactile graphics but often requires that the image author make the graphs manually. The top 3 commercially available applications include: Firebird Graphics Editor by Enabling Technologies, TactileView by Irie AT, and QuickTac by Duxbury Systems. Each of these allow the user to manually draw and edit diagrams and to export to a braille embosser. Only Firebird Graphics Editor allows arbitrary images to be imported while attempting to convert the images automatically. Firebird Graphics Editor by default uses a 2 segment clustering method to convert the graphic but also supports edge detection. When the software separates foreground and background it arbitrarily adds its default texture to one of them but does not leave an outline between regions (as recommended by TVIs). FireBird Graphics Editor is also the only editor that allows the image author to use predefined textures to help distinguish regions. I.1.2 shows examples of FireBird Graphics Editor's automatic image outputs. None of the commercially available application do an adequate job of converting the graphics for a wide range of uses nor do they have enough options or features for most uses, such as: import arbitrary images, outline important components, multiple textures for regions of interest, and the inclusion of ancillary lines to accentuate structures of interest. They also do not attempt to simplify the diagram or remove perspective.


Figure I.1.2: Firebird Graphics Editor in action

## I. 2 Project Focus

## Objectives

1. To explore/modify/develop a variety of automated techniques to segment and simplify an image/drawing into a set of non-fragmented lines/curves that are effective for tactile information processing. Effective will be defined as the ability to convey information about the shape(s) of objects and their parts and/or the identification of objects and their parts, as well as identification of the main spatial relationships between objects and/or object parts in the graphic.
2. To quantitatively compare, for users who are blind or visually impaired and determining by touch alone (those with residual vision will be blindfolded), the effectiveness and ease of use of the two best automated visual-to-tactile conversion methods developed to visual-to-tactile conversions made "by hand" by expert tactile graphic makers. Effectiveness is defined as above and will be assessed by asking questions to the user about the diagram. Ease of use will be defined by the use of the System Usability Scale.
3. To explore/modify/develop a variety of techniques to effectively automatically convert 2D perspective images/drawings into one or more direct views (front, side, top). Effectiveness will be defined as in (1) when examining the direct views.
4. To quantitatively compare, for users who are blind or visually impaired and determining by touch alone (those with residual vision will be blindfolded), the effectiveness and ease of use of the best automated method to segment, simplify and remove perspective to diagrams made "by hand" by expert tactile graphics makers and to the best automated method without the perspective correction. Effectiveness and ease of use are defined as in (1).

Problem Images and diagrams are becoming the sole means for conveying information: approximately $70 \%$ of textbooks convey information solely in diagram form with no text explanations [1]. To use these diagrams people who are blind require sighted individuals to convert visual images to tactile diagrams. The diagram conversion process is quite costly and can take hours to complete, that is to say, when a professional is lucky enough to be available.

Goal This research focuses on the development of an algorithm that can automatically convert visual diagrams and photographs of objects and object scenes into tactile diagrams for people who are blind without any intervention by people who are sighted. This will allow individuals who are blind or visually impaired true independence in accessing tactile diagrams. The focus is on providing information that is difficult to obtain by text descriptions such as the shape of objects and their parts, and the spatial relationships between them. The algorithm will need to work with images found in everyday life such as text book diagrams and store catalogs.

Hypothesis The image processing algorithm designed here to generate tactile diagrams from electronic images will show statistical non-inferiority to images created by professional diagram makers when assessed in terms of their effectiveness in shape interpretation/identification of objects and their basic subparts, and of relationships between objects.

## Deliverables

1. Image processing algorithm that generates tactile diagrams from electronic images with similar user effectiveness and ease of use to using diagrams generated by professional diagram makers.
2. Quantitative results assessing their performance to questions involving shape interpretation/identification of objects and their basic subparts, and of relationships between objects.
3. Set of standard training and testing tactile diagrams for future research.

## I. 3 Preliminary Work

Motivation The motivation for the preliminary work was to explore existing feature spaces and segmentation algorithm spaces in the field of visual image segmentation for the ability to provide effective tactile diagrams. Effective diagrams were defined by a variety of metrics and their comparison to diagrams developed by two TVIs by "hand".

Automatic Image Segmentation Concepts There is a wide variety of existing image segmentation algorithms designed for other applications. All of these algorithms use the similarity of pixels and one or more features (color space, distance metrics, energy, texture, etc.) to determine the segmentation. The different classes considered here represent significantly different approaches to segmentation: edge detection, clustering, graphs, contour detection and differential equations. Several of these algorithms have been expanded in this preliminary work from their original formulation to handle additional features to determine if the added feature space can improve these algorithms for our research purpose.

Assessment Methods In order to compare professionally made diagrams with various image processing algorithms, they need to be converted into a labeled region form such as those in Figure I.3.1. This form allows for four comparisons of congruency between ground truth (professionals) and each algorithm's output: edge, region boundary, region, textured region (multiple non-contiguous sections belonging together). The current standard is to measure matching edge and region boundaries in precision-recall space (PR) with the F-measure. For measuring region matching, the probabilistic rand index is used [52]. Several other metrics including the total run time have been used throughout the literature and are reported here when possible.

Of particular interest is a metric designed to illustrate how the human tactile system would perceive differences in the images called here as the Loomis distance [50]. The Loomis filter is simply a low pass filter that distorts a visual image to appear as the tactile system would understand it. In order to use this as a means to determine how different two images would be perceived, a dissimilarity type distance is measured between the two images after a Loomis filter is applied resulting in the Loomis Distance.


Figure I.3.1: Example tactile graphics from professionals converted to region labels (not to scale).

## I.3.1 Experiment Performed

Three segmentation algorithms were explored across various classes of algorithms using selected sets of representations from the color and texture feature spaces: Kmeans from the clustering class was selected for its speed and the wide range of addons that could be explored later; Level set was selected from the differential equation set because it builds on the others, has better speed/accuracy than other algorithms in the class, and can be easily expanded to include other feature spaces; and Global probability of boundary with oriented watershed ( $\mathrm{gPb}-\mathrm{owt}-\mathrm{ucm}$ ) was selected from the contour detection class because of its recent success over many other segmentation algorithms. Each algorithm was tested over a range of input parameters in order to determine the efficiency of the algorithm and find the best parameter set; the basic parameters and data set are shown in Table I.3.1.

The images used were selected from commonly encountered images in everyday life and school. The images include teaching diagrams, historical landmarks and product advertisements. Images were either photographs or drawn diagrams of single objects or a scene with multiple objects. The images also include common image processing challenges such as: dithering, low print quality, solid color ranges, similar intensities across boundaries, single object with multiple colors, and slow gradients.

Table I.3.1: Experimental constraints

| Data Set | 37 Common Images |  |  |
| :--- | :--- | :--- | :--- |
| Algorithms | K-Mean | Level Set | gPb-owt-ucm |
| Parameters | Color Space | Color Space | UCM Threshold |
|  | $\pm$ Texture Gradient | $\pm$ Texture Gradient | Preset Parameters: |
|  | Distance Metric | Distance Metric | Color Space |
|  | Init Method | Init Method | Color Scales |
|  | Num of Clusters | Num of Phases | Number of Scales |
|  |  | Component Wt | Color Wt |
|  |  | $\Delta$ Component Wt $/ \mathrm{t}$ | Num of Spectra |
|  |  | Init Circle Size | Spectral Wt |
|  |  | Init Circle Padding | Smooth Filter Coefs |
|  |  | Init Circle Shift | Texture Classifier |

## I.3.2 Initial Results and Discussion

The comparison of the time taken and scoring metrics is shown in Table l.3.2. The fastest algorithm was k-means, followed closely by level set and then gPb-owt-ucm. It should be noted that because gPb-owt-ucm has several components that are run outside of Matlab to save time; if those components were implemented in Matlab, it would be considerably slower. Each method was scored based on boundary and region metrics with the current most commonly used metrics being the F-measure and probabilistic rand index (PRI) [41]. Figure I.3.2 shows the results of using PR space to compare the three segmentation algorithms (gPb-owt-ucm, k-mean, level set). This allows for determining the best parameter set as the set closest to the top right corner such as the plots in Figure l.3.2. The figure also shows the points in the trade-off cloud that represent parameters that achieved the best score depending on the metric used. To help determine which parameters are more important, each parameter was statistically examined with a regression model, called "relative" in the figure key. This showed which parameter options would predictably produce good results.

Table I.3.2: Experiment Results: Best score from each algorithm. Professional and novice represent internal consistency whereas each algorithm and novice to professional (N2P) is how well it matches the professional segmentation. Time for all methods represents how long the method took to complete; gPb-owt-ucm, though also implemented in Matlab, is mostly run in C providing a speed boost. The professionals self-reported $>1$ hour more time than shown here for their planning phase before beginning the segmentation and additional time for applying texture to the graphics. Professional measures are based on a mean consistency because all illustrators are expected to have the same level of expertise whereas the novice measurements are based on median to account for extremes.

| Metric | Perfect | Professional | Novice | N2P | K-Means | Level Set | gPb-owt |
| :--- | :--- | :--- | :--- | :--- | :--- | :--- | :--- |
| F | 1.0 | $0.85 \pm 0.09$ | $0.92 \pm 0.08$ | 0.93 | 0.80 | 0.81 | 0.82 |
| D2P | 0.0 |  |  | 0.01 | 0.007 | 0.003 | 0.007 |
| PRI | 1.0 | $0.93 \pm 0.06$ | $0.92 \pm 0.06$ | 0.93 | 0.80 | 0.82 | 0.81 |
| Loomis | 0.0 | $45.4 \pm 22$ | $47.3 \pm 17$ | 45.5 | 68.3 | 66.7 | 62.3 |
| Time | $<1$ Min | 30 Min | 15 Min | 15 Min | 0.2 Sec | 20 Sec | 40 Sec |



Figure I.3.2: PR result points with select points showing the best parameter sets.

From the boundary and region data, four metrics potentially indicate the best segmentation parameter set, namely: the F-measure, distance to professional point in PR space, probabilistic rand index, and Loomis distance. Images created using the best parameters from each of these metrics and from statistical regression (5 total) were exported
and visually analyzed for their similarity to the professionally made images. Through careful analysis, PRI and regression were determined to indicate the best parameters for producing the best images.

## I.3.3 Initial Conclusions

Feature Sets Perceptually based color spaces were anticipated to perform better with the segmentation algorithms than other color spaces and they turned out to be the best. The human visual system processes images in this way on a regular basis so using these color spaces will help a computer see the same thing a person would. However perceptually based color spaces were only marginally better than RGB and there were a few select situations where gray scale performed better. The addition of the texture gradient to the image feature space hindered the ability of all algorithms to segment the images. This is in contrast to past research which showed that the inclusion of texture marginally helps. Therefore, texture will not be included in the feature space for extracting segments of the images.

Algorithms Contour detection with gPb-owt-ucm produced great results in terms of the assessment metrics. It out-performed the other algorithms based on several scoring metrics for segmentation accuracy and consistency. Its limiting factors are: must be run in Linux and it does not produce segments/regions (it only finds boundaries). gPb-owt-ucm is also still far from real time, especially when considering it has already been optimized to avoid Matlab overhead that the other algorithms are still currently bound by. However, the extra time taken may be worthwhile for better performance, which may also shorten the simplification step. Level set performs well enough so far but needs continued work to remove noise. Given that the kMean and level set pixel probability are very similar, the increased noise in level set over kMean is likely due to level set's spatial components, specifically from the gradient. So both gPb-owt-ucm and level set look promising and will
continue to be used in further investigations. kMeans is, unfortunately, too inconsistent and susceptible to noise. If the noise cannot be silenced with some basic techniques it will not be used in the remaining work.

## I. 4 Project Plan

## I.4.1 Design Process

## Development Process



Figure l.4.1: Flow of information in the design process. The process is informed by professional experience and tested with algorithm results against similar diagrams made by professionals.

Steps remaining after completed preliminary work.

1. Segmentation
(a) Finish exploration of segmentation methods by completing further modifications (B1)
(b) Implement noise reduction in level set: clean, merge, split methods (B1)
(c) Implement further noise reduction in level set: spatial distance metrics, gradient metrics, smoothing filters (B1)
(d) Regress gPb -owt boundaries and regions to segments, likely based on color similarity, in order to facilitate addition of texture (B1)
(e) Develop and Implement Automatic texturing scheme, as textured diagrams are more effective than raised line drawings (B1)
(f) Finish digital comparison testing with algorithm updates (C1)
(g) Select 2 best algorithms with their best parameter sets (B2)
(h) Test of segmentation and analysis of its usage (C2), user testing to obtain quantitative and qualitative data to examine algorithm effectiveness in a real tactile exploration task and provide insight for simplification
2. Simplification
(a) Develop simplification algorithm based on application of higher level computer vision algorithms and on insight from the pilot experiment (B3)
(b) Test simplification (C3, Midpoint user testing)
3. Perspective
(a) Develop perspective correction algorithm (B4). Based initially, at least, on parallel/perpendicular lines and texture analysis.
(b) Test full suite (C4, user testing)

## I.4.2 Segmentation

Level set and kMean tend to result in noisy segmentation. With kMean this is caused by its inherent lack of spatial information since it simply groups pixels by color. Level set, however, should produce images with less noise based on the use of spatial components. Unfortunately, the results thus far have actually shown that level set seems to be producing more noise than kMean. Figure I.4.2 shows some of the noise produced by level set which tends to lie in areas of heavy noise in the original image, which likely contributes to false highs in the gradient. This is likely due to noise in the calculated gradient. To help reduce this noise, different gradient measures and smoothing filters will be tested. Possible gradient measures that will be investigated are: central differences, monogenic signals, and multiscale probability of boundaries. Further improvements will be sought through varying the smoothing filter applied to the gradient and weighting each color layer differently. In an additional effort to remove the noise that can interfere with user understanding, the splitting/merging of phases will be explored.


Figure I.4.2: Example of noise left after automatic segmentation using level set.

## I.4.2.1 Automatic Texturing Scheme

Using diagrams that have textured regions significantly improves performance over the use of raised line drawings. Therefore, from the results of segmentation, each diagram will need to have its objects and object parts labeled with different textures. The simplest and, likely, the best method for applying texture will involve assigning texture based on segment area. This would be accomplished by leaving the largest segment, likely the background, without a texture. Then a texture will be selected for each smaller segment until the texture pool is exhausted. The texture pool has been selected from previous research by the Royal National Institute of the Blind. The set consists of five textures that are easily distinguishable from each other. Further understanding will be sought as to how the professionals (TVIs) determine the best texture placement to improve an algorithm's response.

One difficulty with the above method for the global probability of boundary method (gPb-owt-ucm) is that it does not produce a segmented image. Rather it produces an image where each isolated contiguous region has its own label. This is not conducive to applying texture as multiple regions may belong together and need the same texture. For example, petals on a flower should have the same texture and be distinguishable from the textures of the leaves. Regression of these region images to segments will be needed. This will initially be done by matching similar colors between regions. Further understanding on how the professionals decide which regions receive the same texture
will be sought to improve the regression algorithm's response.

## I.4.2.2 User Testing and Analysis using Segmentation Algorithms

One of the most important components of this research is to periodically implement user validation. This will consist of blind or visually impaired users feeling the textured diagrams produced by professionals and those produced by the best of two of the automated methods. There will be one set of images/diagrams that will represent those produced by professional diagram makers. These diagrams are based on separate sets created by each of our two professionals and then reconciled, through discussion, with each other to produce a single set. The aim when the professionals create the diagrams is to represent the "base level" of representation. For example, a diagram of a turtle would show all of its parts but not the details of the shell pattern. The shell pattern would be later shown with an additional diagram. The extraction of this future information will be kept in mind when developing the algorithms but not implemented in the current work. In addition, to the hand-made professional set, there will be two diagram sets representing the two most promising automated methods.

Each set consists of tactile diagrams for 30 visual images and 30 visual diagrams. The images/diagrams are in sets of three on the same topic (e.g. chairs, fruit, etc.) to provide approximately similar difficulty. The images/diagrams are blocked into sets of 10 images and 10 diagrams with one image/diagram from each triplet. A different block of 10 images and 10 diagrams will be provided for each condition (created set). This will avoid bias due to previous exposure to a particular diagram. Blocks will be counterbalanced across subjects between conditions. After each condition (consisting of one set of 10 of images and one set of 10 of diagrams), the system usability survey will be given to access usability of the method. The diagrams will be presented on swell paper.

The users will be asked to identify the object(s) in the image, their general shapes and parts, and their relationships to each other. They will only be told at the beginning of the
experiment that the images they will experience are ones they would experience in school textbooks in history, geography or biology, or in advertisements. The response variables will be whether the users have gotten the questions asked correct and the response time.

For this particular experiment (assessing diagrams automatically created by segmentation only), the results of user testing are intended to provide a baseline understanding of professional image identifiability and show how much simplification is needed after segmentation. Users will be asked to talk through their diagram exploration for each of the diagrams. They will be asked to describe their exploration strategy, what they are feeling and whether it is confusing. Then qualitative content analysis $[72,73]$ will be used to look for themes in what was easy to understand and what was problematic for each of the automated sets in comparison to the professional set. These results will be used to help lead the automatic simplification development. As users will be describing their diagram exploration, for this first experiment only, the response time will not be recorded.

For all experiments, participants will be blind or visually impaired. All participants will be blindfolded to prevent any residual vision from being used as the focus is on nonvisual access. In the future, the use of residual vision will be examined. There will be an attempt to use participants that cover the span of the user population: along the blindness spectrum, congenital versus adventitiously blind, both below and above 65, and with different extents of usage of tactile diagrams. However, as participant recruitment is extremely difficulty, no specific numbers will be aimed for. Individuals will not be included as participants if they have peripheral neuropathy and cannot feel the lines or textures. There will be approximately 7 subjects in this experiment, and 24 subjects in the two remaining experiments.

## Process \& protocol

1. 20 image triplets consisting of 10 picture topics and 10 drawings/diagrams topics each with 3 distinct versions of each image with similar complexity (versions $A, B, C$ ) will be used for a base
2. Each image will be processed (converted to tactile diagram) with 3 methods ( 1 professional, 2 gPb -
owt-ucm, 3 level set)
3. The participant will be told before testing begins where they may encounter such images (school text book, magazine advertisement, etc.).
4. Each participant will be presented with each image in such a way that they receive each version (A/B/C) and each method ( $1 / 2 / 3$ ) once and only once for a particular image blocked and counterbalanced for each diagram set. Each image will be presented on its own swell paper page with the entire image shown on the page.
5. The participants will be allowed to explore the diagrams freely with both hands. In the first experiment they will be asked to describe what they are doing.
6. Participants will be asked to name the object(s) in the diagram, describe its overall shape(s), its parts, and their relationship to each other (answer and time to answer recorded)
7. Participants will be asked to rate the diagram's complexity on a 10 point scale (answer)
8. At the end of each block for each method, the system usability survey will be given.

Statistical Analysis Once the quantitative data is collected, it will be analyzed to determine if the diagrams generated by the automatic algorithms are as identifiable as the diagrams made by professionals. See Appendix I. 6 for list of variables.

1. Generalized linear mixed-effects models will be used
2. The primary variable to be analyzed is correctness of the answers
3. The second variable will be response time
4. Data will be examined primarily for effects due to method
5. Effects due to congenital blindness vs adventitious blindness will be examined (if participant pool size allows for this analysis). Individuals will be defined as congenitally blind if they became blind before the age of two. Individuals will be defined as adventitiously blind if they became blind after the age of 16. [74]

## I.4.3 Simplification

The professional illustrators spend a significant portion of their time determining how to simplify the image. In many cases this is simply a matter of which ancillary (non-region bounding lines) to include and what extraneous information to ignore when producing the tactile diagrams. Figure I. 4.3 shows some examples of diagrams that need lines removed
and those that need lines added back after segmentation. The professionals also spend extra time determining which regions are deserving of textures to enhance user understanding. In other cases, the lines being used are simplified so that the regions represent simpler polygons. This makes it easier to interpret the overall shape of an object or object part. Finally, small objects and/or parts may be removed from the diagram to reduce the complexity of the diagram. This may remove some potentially useful information but the tactile sensory system cannot handle the same amount of information as the visual system. In order for understanding of the basic information provided in a diagram, other pieces of information need to be sacrificed. In some cases, large objects may need to be removed to aid with basic understanding.

In terms of considering ancillary lines, the choice to add or subtract information in an image will likely come down to the overall entropy of the boundary image. Entropy is linked to how much information there is in an image and therefor linked to how well a human would be able to perceive the information in tactile form. If there is too much information then extra lines often represented by weaker probability in the case of global probability of boundary will be removed. Whereas low information will indicate that additional lines need to be added back. These additional lines will come from either higher probability boundaries or high weighted lines from edge detection.


Figure I.4.3: Example of simplification needed. Top row is professionally segmented images while bottom row is computer generated. The fist image has too much information remaining while the latter two are missing information.

Simplification of the object/object part shapes and removal of object/object parts will likely be based on: (a) the assumption that whatever object/objects are closest to the center of the diagram are the most important, and (b) small variations in the boundary of an object/object part (in comparison to the overall variations in that particular boundary) will not be important. Higher level vision algorithms will be considered for this component of diagram simplification.

The results from the qualitative content analysis of the commentary data collected in the first study is expected to provide insight into the issues involving ancillary lines, boundary simplification and removal of objects.

Once the automatic algorithm is further enhanced, new automatic assessment techniques (during the development process prior to user testing) will need to be explored. The new methods will need to take the ancillary lines, whether or not texture should be used, boundary simplification, and object removal into account when determining how well the automatic algorithm matches the professionals. Figure I.4.4 shows the major differences between what is compared after only segmentation and what will need to be compared after further simplification.


Figure l.4.4: Example of needed additions to assessment methods. Each row shows segmentation from different professionals. The first column shows only the outlines. The second column shows the regions that are scored against each other. The third column shows the boundaries that are scored against each other. The forth column shows the complete diagram generated by the professionals.

User Testing of Simplification Methods The user testing of the simplification methods will use the same diagram set as for the previous experiment examining the results from segmentation only. As such, participants in the previous experiment will not be allowed to
participate in this experiment. Again the two best automated techniques will be compared against the set representing the professionals. The two best automated techniques may mean the two best segmentation algorithms with the same simplification technique, the best segmentation algorithm with the two best simplification techniques, or each method being completely independent. The experimental method will be the same as for the previous experiment, except the participants will not be required to talk about what they are doing and response time for all questions will be recorded.

## I.4.4 Perspective

Tactile diagrams that contain perspective are typically very difficult to interpret correctly. Generally, when a user explores a tactile diagram, perspective lines (changes in 3D depth on a 2D surface) are interpreted as a change in size or shape. To ensure the user is not confused by depth information, professional diagram makers usually remove perspective and, when needed, split the diagram into multiple diagrams representing the view from each exhibited side individually. This same functionality needs to be explored in the automatic image conversion algorithm. However, removing perspective can cause problems. Figure I .4 .5 shows how a user might interpret the image and how simple perspective correction can only remove the major perspective while over-exaggerating or distorting minor perspective. As a part of the simplification process, the minor perspectives that become distorted may simply be excluded from the outline drawing by the professionals or they may be adjusted.


Figure l.4.5: Example of perspective in images and how they would be perceived as a shape change compared to an image where the major perspective is removed. Generally the minor perspective would not be included in the outline so the distortion caused to the minor perspective would not harm the user's understanding.

Distortions in images from perspective can manifest in two basic forms, lines that were parallel that are no longer parallel and skew in texture patterns. Lines that were parallel now have a distant intersection point outside of the image boundary called a vanishing point. This vanishing point can be used to figure out how distorted an image has become so that it can be regressed [75-77]. Distortions in texture can be found as shape/size changes, frequency changes, or phase changes. By mapping the textures in the original perspective image the distortions can be found and removed [78, 79]. Both of these methods will be investigated for removing perspective and correcting for distortion.

User Testing of Perspective and Simplification Methods The user testing of the perspective and simplification methods will use a new set of 30 images and 30 diagrams. Again, the images/diagrams will be organized in triplets. This time, all images/diagrams will exhibit perspective. In the conversion process to tactile diagrams two to three diagrams will be created consisting of the front view, the side view and/or the top view. The three sets that will be compared will be slightly different than in previous experiments. Again a set based on those produced by the professionals that have then been reconciled into a single set will be used. For these diagrams, the professionals will be asked to remove the perspective as they normally do when creating tactile diagrams. Another set will be the best automated simplification method (based on the previous testing) with perspective removal into multiple views. The final set will be the best automated simpli-
fication method with no perspective removal. The experimental method will be the same as for the previous experiment.

## I. 5 Budget

## BUDGET JUSTIFICATION

A. SENIOR PERSONNEL -

- Support for Dianne Pawluk, PI, is budgeted at 2.0 summer months, with the required addition of fringe benefits.
- Support for Carolyn Graham, Senior personnel, is budgeted at 1.2 months, with the required addition of firnge benefits.
B. OTHER PERSONNEL
- Graduate Students: student support is requested for one graduate student for all 3 years.
- Undergraduate Students: undergraduate student support will primarily use the REU funding mechanism, however, support is requested specifically for year 1 ( $\$ 10 /$ hour for 20 hours a week during fall and spring terms) for assistance in the experiments doing the picture set equivalency.
- Consultants

Janice Johnson, professional tactile graphics creator/advisory board member
$\$ 3500$ in year $1, \$ 3500$ in year 2 and $\$ 2000$ in year 3
Kit Burnett, professional tactile graphics creator/advisory board member
$\$ 3500$ in year $1, \$ 3500$ in year 2 and $\$ 2000$ in year 3
Both Janice and Kit are experienced teachers of $K-12$ students who are blind and visually impaired; their resumes are included in the CV section. The average consultant rate for teachers of students who are blind and visually impaired is between \$65-\$80 an hour. The number of days of expected service is approximately 40 days and both Janice and Kit have consented to be involved in the grant below their normal pay level.

Barbard McCarthy, advisory board member not allowed to accept payment Barbara is the Director of the Library and Resource Center at the Virginia Department for the Blind and Vision Impaired. Her expertise is in tactile graphics and Braille, and their production using commercial devices.
Joe McNulty, advisory board member $\$ 500$ per year for all 3 years Joe is the Director of the Helen Keller National Center for Deaf-Blind Youths and Adults. His expertise is in disability issues, education and deaf-blind yourths and adults.
Lucia Hasty, advisory board member $\quad \$ 500$ per year for all 3 years
Lucia Hasty is considered a leading expert in the production fo tactile graphics and has been the main contributor to the Braille Authority of North America guidelines for creating tactile diagrams.

The number of days of expected service is approximately 6 days.

## C. FRINGE BENEFITS -

Fringe benefits have been calculated at the rate of $28.0 \%$ for full-time faculty and staff and $8.8 \%$ for summer and hourly personnel for requested salary support for all years of the project based on Virginia Commonwealth University's agreement with DHHS dated 6/10/2011. Full-time students benefit rates equal 0\%.
D. EQUIPMENT - none
E. TRAVEL -

- Conferences - $\$ 6000$ per year for all $\mathbf{3}$ years is needed for travel to research conferences for the PI and students, e.g., the ACM ASSETS Conference on Computers and Accessibility, the IEEE Conference in Engineering in Medicine and Biology, and the International

Conference on Computers Helping People with Special Needs. All conferences are international conferences.
F. PARTICIPANT SUPPORT COSTS - None
G. OTHER DIRECT COSTS -
a. Materials and supplies:

The PI has requested $\$ 2700$ in year 1 for a computer system and 2 copies of Adobe Illustrator (the latter to be supplied to the two professional tactile graphics creators to assist in the creation of tactile diagrams).
$\$ 300$ per year for all 3 years for swell paper.
b. Other:

Stipend for each participant in experimental testing of $\$ 10$ per experiment hour. The number of participants required for each experiment is given below. As transportation to unfamiliar places is very difficult for people who are blind or visually impaired and may prevent them from participating, transportation costs will also be covered for participants (all of whom will be blind or visually impaired); estimated at $\$ 50$ per experiment day.
i. Picture Set Equivalency Experiments:

Experiment a: 10 participants* ( 6 hours* $\$ 10 / \mathrm{hr}+2$ days*50)=\$1600
Experiment b: 10 participants* ( 6 hours ${ }^{*} \$ 10 / h r+2$ days ${ }^{*} 50$ ) $=\$ 1600$.
Funds are requested in year 1.
ii. Experiment 1, 2 days: 24 participants ${ }^{*}\left(6\right.$ hours $^{*} \$ 10 / h r+2$ days $\left.* \$ 50\right)=\$ 3840$. Funds are requested in year 2.
iii. Experiment 2, 2 days: 24 participants* $\left(6\right.$ hours $^{*} \$ 10 / h r+2$ days $\left.{ }^{*} \$ 50\right)=\$ 3840$. Funds are requested in year 3.
iv. Pilot studies: there are expected to be 2 pilot studies for the first experiment (as design decisions are complex) and 1 pilot study for the second experiment. There will be in total 3 pilot studies, approximately 1 per year. 6 participants* $\left(6\right.$ hours $^{*} \$ 10 / \mathrm{hr}+2$ days*50) $=\$ 960$. $\$ 960$ is requested per year for all 3 years.
c. Tuition - Graduate student tuition for one student.

## I. FACILITIES AND ADMINISTRATIVE COSTS

F\&A/Indirect Costs have been calculated on a modified total direct costs base of $49.5 \%$ based on Virginia Commonwealth University's current negotiated rate agreement under DHHS dated 6/10/2011 and administered by Phat, Chau, (301) 492-4855.

## I. 6 Stats Base

| Image (Input) |  |  | User interaction (Output) |  |  |  |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| Code | Description | Comment | Code | Description | Comment | Action | Direct Question |  |
| ID | Drawing image |  | D\# | Difficulty \# | 1-10 |  | How difficult was this image? Explain? |  |
| IP | Picture image |  | F0 | No major fixations | < 10 sec |  |  |  |
| A\# | Algorithm number | 0 - Human | F1 | Fixation on detail 1 | 10-20 sec | Note location |  |  |
| DH | High detail |  | F2 | Fixation on detail 2 | 20-30 sec | Note location |  |  |
| DM | Medium detail |  | F3 | Fixation on detail 3 | $>30 \mathrm{sec}$ | Note location |  |  |
| DL | Low detail |  | HL | Local hand movement | Explored only local area | Note location |  |  |
| DE | Extra detail | Comp to human | HG | Global hand movement | Explored whole image |  |  |  |
| DM | Missing detail | Comp to human | H\# | Number hands used | 1-2 hands |  |  |  |
| NS | Speckle noise |  | I\# | Interaction time \# | 1:0.5:10 min (11 is $>10 \mathrm{~min}$ ) |  |  |  |
| NL | Line noise |  | CC | Char of concussion | Describe area | Note location | What areas of the image were confusing? |  |
| NE | Edge distortions | Comp to human | CS | Char of simple | Describe area | Note location | What areas of the image were easy to understood? |  |
| PS | Perspective stays |  | CF | Char of fixation | Describe area | Note location | Why did you focus on this area? |  |
| PR | Perspective removed |  |  |  |  |  |  |  |
| CS | Smooth contours |  | Characteristics of image areas |  |  |  |  |  |
| CM | Medium/mixed contours |  | Code | Description | Comment | Action |  |  |
| CR | Rough contours |  | CC/S/F - S | Sharp |  |  |  |  |
| TC\# | Texture coverage \# | 0-100\% | CC/S/F - J | Jagged |  |  |  |  |
| T\# | Textures used \# | 0-6 textures | CC/S/F - O | Rounded |  |  |  |  |
|  |  |  | CC/S/F - T\# | Texture | Texture \# |  |  |  |
| User (Input) |  |  | CC/S/F - NT | No Texture |  |  |  |  |
| Code | Description | Comment | CC/S/F - TL | Thin line |  |  |  |  |
| BC | Congenitally blind |  | CC/S/F - BL | Thick line |  |  |  |  |
| BL | Late Blind |  | CC/S/F-C | Closed |  |  |  |  |
|  |  |  | CC/S/F - U | Open |  |  |  |  |
|  |  |  | CC/S/F-S | Smooth |  |  |  |  |
|  |  |  | CC/S/F - R | Rough |  |  |  |  |
|  |  |  | CC/S/F - N | Noisy, Cluttered |  |  |  |  |

## I. 7 Presentation Slides

## VCU Rehabilitation <br> Technology and Haptics Laboratory

Automatic Image Processing and $\qquad$ Conversion to Tactile Graphics

## Tyler Ferro

## 

## Outline

$\qquad$

- Introduction $\qquad$
- Background
- Manual conversion
- Automatic conversion
$\qquad$
- Preliminary Work
- Features and segmentation algorithm classes $\qquad$
- Assessment methods
- Initial Segmentation Results $\qquad$
- Project Plan


## 

## Problem

- ~285 million people - blind or visually $\qquad$ impaired (BVI)
- 12\% (USA) - Bachelors
- 38\% (USA) - Employed
- 32\% (USA) - Below Poverty
- Why are diagrams are important?
- Difficulties in getting diagrams
- Costly support
- Rely on sighted diagram makers

World Health Organization 2014, National Federation of the Blind NFB 2014

[^7]
## Problem

## - Diagram Creation

- Costly Medium - drawing boards, buildup displays, embossed paper, vacuum forming, micro-capsule paper, silk screening, etc
$\qquad$
$\qquad$
$\qquad$

$\qquad$
$\qquad$
$\qquad$
$\qquad$


## Problem

## - Diagram Creation

$\qquad$

- Professional
- Hours to complete
- Limitations of Touch
- Spatial resolution
- Temporal resolution
- Field of View

| Sense Modality | Limit bits/sec |
| :--- | :---: |
| Skin | $10^{2}$ |
| Ear | $10^{4}$ |
| Eye | $10^{6}$ |

$\qquad$
$\qquad$
$\qquad$

- Serial vs Parallel processing


## 

$\qquad$
$\qquad$

## Problem

- Diagram Creation - Design principals address the limitations of touch
- Proportion/scale
- Perspective inclusion/exclusion,
- Simplification
- Removal of extraneous content
- Simplification of shapes


## Motivation

- Lack of availability
- Time consuming process
- Commercially available support
- Software packages
- Limitations
- Past Research
- Narrow focused problem
- Limitations
- Beyond Segmentation
- Simplification - noise, clutter, object specifics, dynamic zooming
- Perspective correction


## (iii) $\mathrm{VCU}_{\text {School of Engineering | Bimendaral Equmenemp }}$

$\qquad$
$\qquad$
$\qquad$
$\qquad$
$\qquad$
$\qquad$
$\qquad$

## Manual

- Created by professionals

$\qquad$
$\qquad$
$\qquad$

Manual Digital Conversion Procedure $\qquad$

- Computer aided graphics $\qquad$
- 2 Professionals
- Textures

$\qquad$
$\qquad$
$\qquad$
$\qquad$
$\qquad$


## Automatic

- Commercial Software - Firebird Graphics Edior



## Automatic

- Ladner et al.
- Text book graphs
- Photoshop plugin


## 

## Automatic

- Barner et al.
- Edge detection, kMean clustering (RGB color space), watershead
- Vector images (diagrams only)
- Limited user testing, constrained questions



## Automatic

- Textured diagrams (Burch2011, Edman1992) $\qquad$
- Segmentation for touch (Way 1997a)
- Removal of details - Limitations of touch (Way1997a, Edman1992)
- Removal of extraneous content - Limitations of touch (Edman1992)
- Removal of perspective - Limitations of touch (Edman1992)


## (iii) $V \mathrm{CU}_{\text {School of Engineering } \mid \text { Bionedical E Enineaing }}$

## Preliminary Work

## Manual Digital Conversion Procedure

- Texture

$\qquad$
$\qquad$
$\qquad$
$\qquad$
$\qquad$
$\qquad$
$\qquad$


## Segmentation

- Considerations of current algorithms: $\qquad$
- Feature spaces
- Algorithms $\qquad$
- Consideration of assessment methods
- Comparing results to previous work
- Relevance for tactile graphics


## Feature Space

## - Color

- Basic: Gray, RGB, CYMK, HSL
- Perceptual: Luv, Lab

$\qquad$
$\qquad$
$\qquad$
$\qquad$
$\qquad$
$\qquad$

$\qquad$


## Example Color Space


$\qquad$
$\qquad$
$\qquad$
$\qquad$
$\qquad$

## Example Color Space


$\qquad$
$\qquad$
$\qquad$
$\qquad$
$\qquad$
$\qquad$

## Example Color Space

$\qquad$
$\qquad$
$\qquad$
$\qquad$
$\qquad$
$\qquad$
$\qquad$

Feature Space

- Energy (Gradient) - Roberts, Prewit, Sobel, , Kiscc, Lapacaian of Gaussian, SUSAN, Phase Congruency, Monogenic Signal, Oriented Energies, Oriented Gradient of Histograms, Probability of boundary
- Orientation - Robers, Prewitt, Sobel, Kisch, Phase Congruency. Monogenic Signal, Oriented Energies, Oriented Gradient of Histograms
- Phase - Phase Congruency, Monogenic Signal



## Feature Space

- Texture - texture gradient, texture classification
- Distance - Euclid, Dihedral, Normalized, Mahal
- Spectral - Eigen values of affinity matrix



## Segmentation Classes

- Edge Detection
- Clustering
- Graphing
- Contour Detection
- Differential Equations


## Edge Detection

- Energy Threshold - single, Hysteresis
- Algorithms - Any energy operator
- Limitation - Non-continuous boundaries, cannot handle subjective contours, large impact from noise


## Clustering

- Group based on similarity - Generally only color $\qquad$
- Algorithms - к-Means, Mean Shitt, DBSCAN, Spectral, DPGMm.
- Limitation - Computationally expensive (CPU time, except K Mean), Works on continuous dimensions only, cannot handle subjective contours

$\qquad$
$\qquad$
$\qquad$
$\qquad$


## Graphing

- Treat pixel or pixel group as node $\qquad$ connected by edges
- Cut edges - between dissimilar nodes or across high energy
$\qquad$
- Algorithms - Min-Cuts, Normalized Cuts, SE Min-Cuts. $\qquad$
- Limitation - Computationally expensive (CPU time), cannot handle subjective contours

Low contrast strong coupling

- High contrast weak coupling

$\qquad$
$\qquad$

Shi2004

## Contour Detection

- Topography $\qquad$
- Algorithms - Watershed on energy or probability of boundary (gPb-owt-ucm)
- Limitation - Best is partially obfuscated and proprietary, cannot handle subjective contours, no grouping of like pixels for

$\qquad$
$\qquad$
$\qquad$


## Differential Equations

- Propagation - Region contour around object $\qquad$
- Algorithms - Region growing, Active contour, Level set
- Limitation - Will not cross boundary (except Level Set), foreground/background/single-object only (except Level Set)

$\qquad$
$\qquad$
$\qquad$
$\qquad$

$\qquad$


## Assessment Methods


$\qquad$
$\qquad$
$\qquad$
$\qquad$
$\qquad$
$\qquad$

## Assessment Methods

- Boundary - F-Score/F-Measure, Distance to Human

$\qquad$
$\qquad$
$\qquad$
$\qquad$
$\qquad$
$\qquad$
$\qquad$

Assessment Methods $\qquad$

- Region/Information - Probabilistic Rand Index $\qquad$
- Perceptual Boundary - Loomis Distance

$\qquad$
$\qquad$
$\qquad$
$\qquad$
$\qquad$

Pro vs. Novice vs. BSDS $\qquad$

$\qquad$
$\qquad$
$\qquad$
$\qquad$
$\qquad$
$\qquad$

Pro vs. Novice

(iii) $V \mathrm{CU}_{\text {School of Engineering } \mid \text { I Bionedical Evinneering }}$

Preliminary Work

## Pro vs. Novice

- Temporal trend? No - 0.1\% Slope



## Initial Work

- Parameter Sets Examined

| Data Set | 37 Common Images |  |  |
| :--- | :--- | :--- | :--- |
| Algorithms | K-Mean | Level Set | gPb-owt-ucm |
| Parameters | Color Space | Color Space | UCM Threshold |
|  | $\pm$ Texture Gradient | $\pm$ Texture Gradient | Preset Parameters: |
|  | Distance Metric | Distance Metric | Color Space |
|  | Init Method | Init Method | Color Scales |
|  | Num of Clusters | Num of Phases | Number of Scales |
|  |  | Component Wt | Color Wt |
|  |  | $\Delta$ Component Wt/t | Num of Spectra |
|  |  | Init Circle Size | Spectral Wt |
|  |  | Init Circle Padding | Smooth Filter Coefs |
|  |  | Init Circle Shift | Texture Classifier |



## Initial Results

- PR (Precision-Recall) space

$\qquad$
$\qquad$
$\qquad$
$\qquad$
$\qquad$
$\qquad$
$\qquad$


## Initial Results

$\qquad$

- PR curves

$\qquad$
$\qquad$
$\qquad$
$\qquad$
$\qquad$

Initial Results

- F-Score, Distance to Professional (D2P), $\qquad$ Probabilistic Rand Index (PRI), Loomis Distance, Time


Preliminary Work


## Initial Results

- Statistical regression models


Preliminary Work


## Initial Results

- Which measure is best?
- Why is distance to professional better than F?
- Could better results be hidden due to pixel correspondence?

| Description | Mean | Median |
| :--- | ---: | ---: |
| Regression | 1.92 | 1.6 |
| PRI | 1.93 | 1.6 |
| F | 2.26 | 2.2 |
| LD | 2.29 | 2.2 |
| D2P | 2.58 | 2.8 |

(1iii) Preliminary Work

## Initial Results

- Which measure is best? $\qquad$
$\qquad$
$\qquad$
$\qquad$
$\qquad$
$\qquad$
$\qquad$
$\qquad$
$\qquad$
$\qquad$
$\qquad$
$\qquad$
$\qquad$
$\qquad$
$\qquad$
$\qquad$
$\qquad$
$\qquad$
$\qquad$
$\qquad$



## Initial Results

- Consistency
- kMean
- F - 0.86-0.99 (med 0.99)
- PRI - 0.72-0.98 (med 0.93)
- LevelSet
- F-0.76-1 (med 1)
- PRI-0.96-1 (med 1)
- gPb-owt-ucm
- F/PRI - 1


##  <br> Preliminary Work

## Initial Conclusions

$\qquad$

- Feature Sets
- Perceptually based colors are often better
- Texture gradient hinders performance


## Initial Conclusions

- Algorithms
- Contour detection is great, but needs work
- LevelSet is ok but, needs work
- kMean is less consistent and more susceptible to noise
$\qquad$


## Initial Conclusions

## - Assessment Methods

##  <br> Preliminary Work

## Initial Conclusions

- Professional's simplification (level of detail) is influenced by experience


## (iii) $\mathrm{VCU}_{\text {School of Engineering } \mid \text { Bionderal Enjineating }}$

## Project Plan

$\qquad$

- Implement/Develop
- Noise reduction: split/merge on LevelSet and spatial constraints on LevelSet
- Regress gPb-owt boundaries
- Automatic texturing scheme
- Preliminary user testing - Qualitative content analysis
- Implement/Develop
- Simplification algorithm
- Scoring scheme - include ancillary lines and textured region analysis
- Comparison experiment
- Perspective correction
- Final comparison experiment

Noise Reduction (LevelSet)

- Split Segments
- Merge Segments
- Clean Small Segments
- Spatial Acuity

$\qquad$
$\qquad$
$\qquad$
$\qquad$


## Simplification

- Remove extraneous information $\qquad$
- Add ancillary lines

$\qquad$
$\qquad$
$\qquad$
$\qquad$
$\qquad$


## Assessment Methods


$\qquad$
$\qquad$
$\qquad$
$\qquad$
$\qquad$
$\qquad$

## Assessment Methods

- Account for ancillary lines $\qquad$
- Account for texture regions

$\qquad$
$\qquad$
$\qquad$
$\qquad$
$\qquad$


## Perspective Correction

$\qquad$

- Presentation angle of an image $\qquad$
- Users perceive this as an odd shape or size change
$\qquad$
- Approach

$\qquad$
$\qquad$
$\qquad$
$\qquad$


## Acknowledgments

- VCU Biomedical Engineering Dept.
- Advisory committee
- NSF IIS Grant \#1218310


## Appendix J: Literature Review

## J. 1 Features

## J.1.1 Color Space

Images can be stored and processed in many different color spaces. Printers for instance work in Cyan/Yellow/Magenta/Black (CYMK) color space in order to print pigments. Figures J.1.2 and J.2.12 show how color can be seen as either a dimension in color space or a layer of the image. In order to model image processing after the human visual system three color spaces are used throughout this document (Gray, Red/Green/Blue (RGB), and Luv). Gray is the simplest color space as it only has a single dimension representing the brightness of the image. RGB is the most common and intuitive color space for working with images because the human visual system first recognizes color through the RGB color receptors in the retina. Luv/Lab color spaces are based on the combination of brightness with two opposing color dimensions (green vs red and blue vs yellow). Luv/Lab were initially designed to increment based on human perception of color. Figure J.1.3 shows histograms of a uniform color (RGB) distribution image split between RGB and Luv color spaces. The deformed nature of Luv space comes from its basis in human perception causing a nonlinear conversion mapping from RGB to Luv space. Figure J.1.6 shows a newer understanding of the human visual system whereby the initial RGB color sense is converted to brightness and opposing color distribution before leaving the retina. Since users process images ultimately using Luv space in order to gather meaning from images it stands to reason that a computer could also get more of a human understand of images when processing in Luv space. Principal component analysis (PCA) can also be used on a image by image basis to help intensify less dominate features but also runs the rick of over emphasizing unimportant features.

Logically it makes sense to model as much of the system as possible after human vision and perception since that is the same system that naturally processes images. One area that has been particularly considered in this regard are the color spaces. Gray and red/green/blue (RGB) are obvious choices because of their ubiquity. However they are not necessarily based on human perception. Lab/Luv color spaces (brightness, redgreen, blue-yellow) were originally designed to best fit human perception with each increment matching to the same unit step in human visual perception [24]. Around the same time as their creation research determined that human vision seems to convert tri-chromatic (RGB) into opponent colors (Lab/Luv) within the retina before sending the information to the brain [80,81]; This phenomenon is shown in an example in Figure J.1.6. Throughout this research four color spaces are used gray/RGB for their ubiquity and Lu$\mathrm{v} / \mathrm{Lab}$ for their base in human perception in an effort to model segmentation after the human visual system.

Though Lab and Luv are both based in human perception they differ on a few points. Luv color space attempts to account for how different colors at the same brightness appear to have different brightnesses known as the HelmholtzâĂŞKohlrausch effect [82]. Conversely each color's maximum chroma is at different intensity levels; both are shown in Figure J.1.4. Human color perception is much deeper though as perception is also based on a person's surroundings. The human visual system will shift colors based on the general lighting color of a room (examples in Figure J.1.5); in computer color spaces a known white point must be used to help with true color conversions. Lab space uses the Von Kries transform while Luv uses the Judd-type transform [83]. No clear winner came from the debate of which transform was better so both Lab and Luv were made standards at the same time.


Figure J.1.1: Shows various color spaces [24]


Figure J.1.2: Shows various color space layers. The $u$ and $v$ layers in LUV lack brightness information but are shown here with and without brightness to exaggerate the chroma difference of the layers.


Figure J.1.3: Shows RGB and Luv color spaces


Figure J.1.4: Helmholtz-Kohlrausch effect: Each block has the same luminescence (gray scale value) as the background but appear to have different brightnesses. Each color's maximum chroma point is at a different intensity. [82]


Figure J.1.5: Examples of how the surroundings alter color perception. On the left of the figure is an example showing a common issue with images where sun light gives a yellow hue and florescent lights give a blue hue. It also shows how the human visual system becomes tolerant of these background colors. By staring at the black dot for 20 sec then looking at the plane, the overlaying hue seems to disappear. [84] The right side of the figure shows a recent internet phenomenon that illustrates how background color alters the perceived color of the dress. [85]


Figure J.1.6: Shows human color vision pathway and the conversion from trichromatic to opponent [80,81]

## J.1.2 Basic Operators

The following basic operators are used throughout many of the image processing algorithms. Many of these operators represent a distance in either image space or color space that can be used to extract further features. Smoothing filters are also listed here as they are often used throughout image processing at various stages of feature extrac-
tion.

## Distance from point to line

$$
\begin{align*}
t & =\frac{\left(v_{1}-v_{2}\right) \cdot\left(p-v_{2}\right)}{\left(v_{1}-v_{2}\right) \cdot\left(v_{1}-v_{2}\right)} \\
d & =\left(p-v_{2}\right)-t *\left(v_{1}-v_{2}\right)  \tag{J.1.1}\\
d & =\frac{\left\|\left(v_{1}-v_{2}\right) \times\left(p-v_{2}\right)\right\|}{\left\|v_{1}-v_{2}\right\|}
\end{align*}
$$

## Distance from point to plane

$$
\begin{equation*}
d=\|p \cdot \nu\| \tag{J.1.2}
\end{equation*}
$$

Contour curvature

$$
\begin{equation*}
\operatorname{curv}=\left|\frac{f_{x x}\left(1+f_{y}^{2}\right)-2 f_{x y} f_{x} f_{y}+f_{y y}\left(1+f_{x}^{2}\right)}{\left(f_{x}^{2}+f_{y}^{2}+1\right)^{3 / 2}}\right| \tag{J.1.3}
\end{equation*}
$$

## Euclidean distance between points

$$
\begin{equation*}
d=\left\|p_{1}-p_{2}\right\| \tag{J.1.4}
\end{equation*}
$$

Normalized Euclidean Distance normalized by standard deviation (aka Standardization) [48]

$$
\begin{equation*}
\hat{d}=\frac{p-\mu}{\sigma} \tag{J.1.5}
\end{equation*}
$$

Mahalanobis Distance accounts for covariance and independence between dimensions

$$
\begin{equation*}
d_{i}=\left(\left(p_{i}-\mu\right)^{T} S^{-1}(p-\mu)\right) \tag{J.1.6}
\end{equation*}
$$

Arc Angle between two 3D vectors (dihedral) [86]

$$
\begin{equation*}
\Theta=\arccos \frac{p_{1} \cdot p_{2}}{\left\|p_{1}\right\|\left\|p_{2}\right\|} \tag{J.1.7}
\end{equation*}
$$

Chi Squared Distance is used to determine the difference between histograms [87]

$$
\begin{equation*}
\chi^{2}(g, h)=\frac{1}{2} \sum_{i} \frac{\left(x_{i}-y_{i}\right)^{2}}{x_{i}+y_{i}} \tag{J.1.8}
\end{equation*}
$$

Gaussian Kernel [48,88-91]

$$
\begin{equation*}
g=\frac{1}{2 \pi \sigma^{2}} e^{\frac{-\left(x^{2}+y^{2}\right)}{2 \sigma^{2}}} \tag{J.1.9}
\end{equation*}
$$



Figure J.1.7: Gaussian filter

## Error Function (ERF)

$$
\begin{equation*}
\operatorname{erf}(x)=\frac{1}{\sqrt{\pi}} \int_{-x}^{x} e^{-t^{2}} d t \tag{J.1.10}
\end{equation*}
$$

RichardsonâĂŞLucy deconvolution [67]

## J.1.3 Basic Quantifiers

These methods for quantifying and/or re-representing image information are used throughout image processing on color space or image space in order to better extract
higher level features.

Image histograms show the frequency at which a particular pixel intensity happens.
This can be expanded to work with color images to show the frequency each color combination happens. Figures J.1.8 to J.1.10 show both gray scale and color histograms of various images.


Figure J.1.8: Histogram of smiley face


Figure J.1.9: Histogram of Roman aqueduct


Figure J.1.10: Histogram of hot air balloons

Principal Component Analysis (PCA) is a method of determining new axes based on maximizing variation. This method can help intensify less dominate features but also runs the rick of over emphasizing unimportant features. The most common methods for determining the principal axes are Eigenanalysis of the covariance matrix or singular value decomposition (SVD).


Figure J.1.11: Histogram of smiley face with principal axes and plane.

(a) Aqueduct

(c) Histogram with principal plane

(b) Histogram with principal axes

(d) Histogram in PCA space

Figure J.1.12: Histogram of aqueduct with principal axes and plane.


Figure J.1.13: Histogram of balloon with principal axes and plane.

Empirical Mode Decomposition (EMD) This algorithm breaks a signal apart into rough like frequency bins using splines. [92-97]



STEP3. Calculate the mean of the maxima and minima splines


STEP 4. Subtract the mean from the raw signal i.e.1st IMF





vornumms


STEP 2.

STEP 3.

(a) EMD on EEG with step explaination. [92]

(b) EMD is used here to find wonder in an ECG signal.

Figure J.1.14: Examples of EMD


Figure J.1.15: EMD on balloon Image


Figure J.1.16: EMD on aqueduct Image

## J.1.4 First Order Local Energy (Color Gradient)

These kernels are convoluted over the images in order to extract gradients (energy) and orientation. In order for these to work the data must be based on a continuum so that it can extract gradient based on change in a particular dimension (ie. single color). These measures result in a graded potential with higher values representing a greater change in color. Gradient is analogues to measuring the distance or change in color space between adjacent local elements in image space.

$$
\begin{align*}
& 1 \\
& K_{0}=\begin{array}{lll}
1 & 0 & -1
\end{array} K_{1}=0  \tag{J.1.11}\\
& -1 \\
& E=\left(\sum_{0 \leq i \leq n-1}\left(K_{i} * I_{x, y}\right)^{2}\right)^{1 / 2}  \tag{J.1.12}\\
& \Theta=\arctan \left(\frac{K_{1} * I_{x, y}}{K_{2} * I_{x, y}}\right) \tag{J.1.13}
\end{align*}
$$

Roberts Operator [89, 98]

$$
\begin{gather*}
K_{0}=\begin{array}{rr}
1 & 0 \\
0 & -1
\end{array} \quad K_{1}=\begin{array}{rr}
0 & 1 \\
-1 & 0
\end{array}  \tag{J.1.14}\\
E=\max _{0 \leq i \leq n-1}\left\{K_{i} * I_{x, y}\right\} \tag{J.1.15}
\end{gather*}
$$

Prewitt Operator [89, 98, 99]

$$
\begin{gather*}
K_{0}=\begin{array}{rrrrr}
1 & 0 & -1 & 1 & 1 \\
1 & 0 & -1 & K_{1}= & 0 \\
1 & 0 & -1 & 0 \\
E= & -1 & -1 & -1 \\
E=\left(\sum_{0 \leq i \leq n-1}\left(K_{i} * I_{x, y}\right)^{2}\right)^{1 / 2} \\
\Theta=\arctan \left(\frac{K_{1} * I_{x, y}}{K_{2} * I_{x, y}}\right)
\end{array} .
\end{gather*}
$$

$$
\begin{gather*}
K_{0}=\begin{array}{rlrrr}
1 & 0 & -1 & 1 & 2 \\
2 & 0 & -2 & K_{1}= & 0 \\
1 & 0 & -1 & 0 \\
-1 & -2 & -1 \\
E= & \left(\sum_{0 \leq i \leq n-1}\left(K_{i} * I_{x, y}\right)^{2}\right)^{1 / 2} \\
\Theta=\arctan \left(\frac{K_{1} * I_{x, y}}{K_{2} * I_{x, y}}\right)
\end{array} \tag{J.1.19}
\end{gather*}
$$

Kirsch Operator [90, 98, 100]

$$
\begin{align*}
& \begin{array}{lllllllll}
5 & 5 & -3 & 5 & -3 & -3 & -3 & -3 & -3
\end{array} \\
& K_{3}=\begin{array}{llllllllllll}
5 & 0 & -3 & K_{4}= & 5 & 0 & -3 & K_{5}= & 5 & 0 & -3
\end{array}  \tag{J.1.22}\\
& \begin{array}{lllllllll}
-3 & -3 & -3 & 5 & -3 & -3 & 5 & 5 & -3
\end{array} \\
& \begin{array}{llllll}
-3 & -3 & -3 & -3 & -3 & -3
\end{array} \\
& K_{6}=-3 \quad 0 \quad-3 \quad K_{7}=-3 \quad 0 \quad 5 \\
& \begin{array}{llllll}
5 & 5 & 5 & -3 & 5 & 5
\end{array} \\
& E=\max _{0 \leq i \leq n-1}\left\{K_{i} * I_{x, y}\right\}  \tag{J.1.23}\\
& \Theta=\underset{0 \leq i \leq n-1}{\operatorname{Arg} \max }\left\{K_{i} * I_{x, y}\right\} \cdot \pi / 4 \tag{J.1.24}
\end{align*}
$$



Figure J.1.17: Comparison of first order and simple local energy operations applied to image of smiley face. Color space left to right: Gray, RGB, Luv.


Figure J.1.18: Comparison of first order and simple local energy operations used to find orientation. Color space left to right: Gray, RGB, Luv.


Figure J.1.19: Comparison of first order and simple local energy operations applied to image of Roman aqueduct. Color space left to right: Gray, RGB, Luv.


Figure J.1.20: Comparison of first order and simple local energy operations applied to image of hot air balloons. Color space left to right: Gray, RGB, Luv.

## Compare

## J.1.5 Second Order Local Energy

These kernels are convoluted over the images in order to extract gradients (energy) and cannot retrieve orientation. In order for these to work the data must be based on a continuum so that it can extract gradient based on change in a particular dimension (ie. single color). These measures result in an indication only if the data is at an inflection point (zero cross of 2nd derivative) but cannot indicate the rate of change.

Laplacian [89,98]

$$
\begin{align*}
& \nabla^{2} I_{x, y}=\frac{\partial^{2}}{\partial x^{2}} I_{x, y}+\frac{\partial^{2}}{\partial y^{2}} I_{x, y}  \tag{J.1.25}\\
& \begin{array}{llllll}
0 & -1 & 0 & -1 & -1 & -1
\end{array} \\
& K=-1 \quad 4 \quad-1 \quad \text { or } \quad K^{\prime}=-1 \quad 8 \quad-1  \tag{J.1.26}\\
& \begin{array}{llllll}
0 & -1 & 0 & -1 & -1 & -1
\end{array} \\
& E=\text { Zerocross }\left\{I_{x, y} * K\right\} \tag{J.1.27}
\end{align*}
$$

Laplacian of Gaussian [89-91,101]

$$
\begin{gather*}
L o G=\nabla^{2} g=\frac{1}{\sigma}\left(\frac{\left(x^{2}+y^{2}\right)}{\sigma^{2}}-2\right) e^{\frac{-\left(x^{2}+y^{2}\right)}{2 \sigma^{2}}}  \tag{J.1.28}\\
\sigma \nabla^{2} g=\frac{\partial g}{\partial \sigma} \approx \frac{g_{k \sigma}-g_{\sigma}}{k \sigma-\sigma} \tag{J.1.29}
\end{gather*}
$$



Figure J.1.21: Laplacian of Gaussian filter


Figure J.1.22: Comparison of second order local energy operations applied to image of smiley face. Color space left to right: Gray, RGB, Luv.

## J.1.6 Complex Local Energy

These operators are used to extract gradients (energy), orientation, and phase. In order for many (not all) of these to work the data must be based on a continuum so that it can extract gradient based on change in a particular dimension (ie. single color). These measures result in a graded potential with higher values representing a greater change in color.

Oriented Energy is the process of filtering the image with oblong even/odd filters placed at different angles. These filters are similar to a Gabor filter which is thought to be similar to human visual processing. [68, 102, 103]

$$
\begin{gather*}
f_{e}=g_{\sigma_{1}}^{\prime \prime}(y) g_{\sigma_{2}}(x) \\
f_{e}=\frac{d^{2}}{d y^{2}}\left(\frac{1}{c} e^{\left(\frac{y^{2}}{\sigma^{2}}\right)} e^{\left(\frac{x^{2}}{l^{2} \sigma^{2}}\right)}\right)  \tag{J.1.30}\\
f_{o}=H\left(f_{1}^{e}\right)
\end{gather*}
$$



Figure J.1.23: Oriented energy filters

$$
\begin{gather*}
O E_{0^{\circ}}=\left(I_{x, y} * f_{e}\right)^{2}+\left(I_{x, y} * f_{o}\right)^{2} \\
E=\max _{0 \leq \theta<\pi}\left\{O E_{\theta}\right\}  \tag{J.1.31}\\
\Theta=\underset{0 \leq \theta<\pi}{\operatorname{Arg} \max }\left\{O E_{\theta}\right\}
\end{gather*}
$$

Monogenic Signal is the 2d equivalent of 1d analytic signal. This can also be expanded to use quadrature pairs similar to Oriented Energy (even/odd function pair) [104,

104-108]

$$
\begin{gather*}
b=g_{1}-g_{2} \\
h_{1}=\frac{x}{2 \pi\left(x^{2}+y^{2}\right)^{3 / 2}} \\
H_{1}=i \frac{u}{\left(u^{2}+v^{2}\right)^{1 / 2}}  \tag{J.1.32}\\
h_{2}=\frac{y}{2 \pi\left(x^{2}+y^{2}\right)^{3 / 2}} \\
H_{2}=i \frac{v}{\left(u^{2}+v^{2}\right)^{1 / 2}}
\end{gather*}
$$



Figure J.1.24: Monogenic signal filters

$$
\begin{gather*}
\hat{I}=I_{x, y} * b_{x, y} \\
E=\left(\hat{I}^{2}+\left(\hat{I} * h_{1}\right)^{2}+\left(\hat{I} * h_{2}\right)^{2}\right)^{1 / 2} \\
\Theta=\arctan \left(\frac{\hat{I} * h_{1}}{\hat{I} * h_{2}}\right)  \tag{J.1.33}\\
\Phi=\arctan \left(\frac{\hat{I}}{\left(\left(\hat{I} * h_{1}\right)^{2}+\left(\hat{I} * h_{2}\right)^{2}\right)^{1 / 2}}\right)
\end{gather*}
$$

Phase Congruency is the understanding that an edge is more likely to be at a location where all frequency components are in phase. This method attempts to find locations where phases are in sync. [109-111]

(a) Odd, edge

(b) Even, ridge

Figure J.1.25: Phase Congruency

$$
\begin{gather*}
P C=\frac{\|E\|}{\sum_{n} A_{n}} \\
a=\sum(P C(\theta) \cos (\theta))^{2} \\
b=\sum(P C(\theta) \cos (\theta) P C(\theta) \sin (\theta))  \tag{J.1.34}\\
c=\sum(P C(\theta) \sin (\theta))^{2} \\
\Theta=\frac{1}{2} \arctan 2\left(\frac{b}{\sqrt{b^{2}+(a-c)^{2}}}, \frac{a-c}{\sqrt{b^{2}+(a-c)^{2}}}\right)
\end{gather*}
$$

SUSAN Filter is short for smallest univalue segment assimilating nucleus (SUSAN). In a given circle it calculates the number of pixels that are different beyond a threshold from the center pixel. Generally this filter is not affected by noise because of dual thresholds. [112-115]


Figure J.1.26: Susan Filter [114]

## Histogram of Oriented Gradients (HOG) <br> [116]

Oriented Gradient of Histograms is the only filter discussed thus far that does not require its data to be on a continuum. This means that the data can simply be a set of labels and this filter will still represent the amount of difference in a given region with a specific orientation. To help reduce noise and improve signal a Savitzky-Golay filter can be applied. [28,68, 87$]$


Figure J.1.27: Oriented Gradient of Histograms [87]

$$
\begin{gather*}
G_{\theta}=\chi^{2}(g, h)=\frac{1}{2} \sum_{i} \frac{\left(g_{i}-h_{i}\right)^{2}}{g_{i}+h_{i}} \\
E=\max _{0 \leq \theta<\pi} G_{\theta}  \tag{J.1.35}\\
\Theta=\underset{0 \leq \theta<\pi}{\operatorname{Arg} \max }\left\{G_{\theta}\right\}
\end{gather*}
$$



Figure J.1.28: Comparison of complex local energy operations applied to image of smiley face. Color space left to right: Gray, RGB, Luv.


Figure J.1.29: Comparison of complex local energy operations used to find orientation $(\Theta)$. Color space left to right: Gray, RGB, Luv.


Figure J.1.30: Comparison of complex local energy operations used to find phase ( $\Phi$ ). Color space left to right: Gray, RGB, Luv.


Figure J.1.31: Comparison of complex local energy operations applied to image of Roman aqueduct. Color space left to right: Gray, RGB, Luv.


Figure J.1.32: Comparison of complex local energy operations applied to image of hot air balloons. Color space left to right: Gray, RGB, Luv.

## Compare

## J.1.7 Texture

Texture comes from repeated patters of energy or gradient.

Empirical Mode Decomposition (EMD) extracts texture similar to a high pass filter [92-95, 117, 118].

Wavelets extract specific textures using pre-determined filters for each texture of interest [119].

## Steerable filter

Oriented energy [103]

Textons are specific textures similar wavelets but are intrinsic to a particular image and thus do not need to be predefined. These are extracted using oriented energy and k-mean clustering. K-Mean clustering is discussed later for its ability to work as a segmentation method but the same principal is applied here to group like textures in an image. [28, 68, 102, 120]

Local Frequency is the process of extracting texture as similarities in local frequency.


Figure J.1.33: Comparison of select texture operations.

## J.1.8 Texture Gradient

Generally this is a measure of areas with high or low amounts of texture but can also be a measure of discontinuities in texture type. This is similar to color gradient, in place of changing color there is now changing texture.

Smallest Univalue Segment Assimilating Nucleus (SUSAN) This method uses the SUSAN filter to extract information about how much texture is in a given area. Due to the SUSAN filters dual shareholding small changes associated with noise are ignored. [37, 112, 113]

Local Entropy is the process of quantifying the amount of chaos within small areas. This is built into matlab as entropyfilt. This method is highly affected by noise. [27]

Third Moment is quantifying the amount of repetition or pattern in a local area. [27]

Oriented Gradient of Histograms This is process of taking texture labels such as those from wavelets or textons and finding oriented gradient of histograms to determine where a change in texture type occurs [87].


Figure J.1.34: Comparison of texture gradient operations. Color space left to right: Gray, RGB, Luv.


Figure J.1.35: Comparison of texture gradient operations. Color space left to right: Gray, RGB, Luv.

## J.1.9 Texture \& Noise Removal

In many aspects of image processing noise (insignificant texture) needs to be removed or dampened.

Gaussian (Isotropic Diffusion) filter simply blurs everything resulting in dampened noise.

Anisotropic Diffusion is similar to Gaussian filter in that it blurs the image but is is selective and attempts to retain boundaries [48, 121]. Many of the following filters can be considered types of anisotropic diffusion.

$$
\begin{gather*}
\frac{\partial I}{\partial t}=\nabla \cdot g(x, y, t) \nabla I \\
g(x, y, t)=\frac{1}{1+\|\nabla I(x, y, t)\|^{2} / \lambda^{2}}  \tag{J.1.36}\\
\frac{\partial I}{\partial t}=\nabla \cdot D \nabla I \tag{J.1.37}
\end{gather*}
$$



Figure J.1.36: Comparison of 2 Anisotropic Diffusion techniques (bottom) to Gaussian filter (top-right) on a noisy image (top-left) [48].

B-Spline \& Thine Plate Spline are methods of creating smooth line interpolation that cane also be used to smooth data [122-125].

Midrange filter works by finding the middle value in a kernel window. This can be implemented with morphological operators. [126]

$$
\begin{align*}
\operatorname{Mid}(a, b, c) & =\frac{1}{2}(\min (a, b, c)+\max (a, b, c))  \tag{J.1.38}\\
\operatorname{Mid} & =\frac{1}{2}(\text { Erosion }+ \text { Dilation })
\end{align*}
$$

Median \& Pseudomedian (PM) work by finding the median value in a kernel window. Pseudomedian filter attempts to come close to true median but has a faster implementation for larger sets because it only requires a sliding sequence as opposed to all combina-
tions. Pseudomedian filter can also be implemented as a combination of morphological operators (open/close). [126, 127]

$$
\begin{gather*}
\operatorname{Med}(a, b, c)=\max (\min (a, b), \min (b, c), \min (a, c))  \tag{J.1.39}\\
\operatorname{Med}(a, b, c)=\min (\max (a, b), \max (b, c), \max (a, c)) \\
p M e d(a, b, c)=\frac{1}{2} \max (\min (a, b), \min (b, c))+\frac{1}{2} \min (\max (a, b), \max (b, c))  \tag{J.1.40}\\
p M e d=\frac{1}{2} \text { Open }+\frac{1}{2} C l o s e
\end{gather*}
$$

## Linear Combination of open-closing and close-opening (LOCO)

$$
\begin{equation*}
L O C O=\frac{1}{2} \text { OpenClose }+\frac{1}{2} \text { CloseOpen } \tag{J.1.41}
\end{equation*}
$$

Gradient Inverse Weighted (GIW) filter smooths areas of lower gradient more than areas of higher gradient. Multiple iterations will look like median filter. [128]

Wiener filter [129]

Empirical Mode Decomposition (EMD) can be used like a high pass filter to removed noise [92-95]..

Smallest Univalue Segment Assimilating Nucleus (SUSAN) method uses the SUSAN filter to locate noise then blur around it while ignoring larger changes like edges or boundaries. Multiple iterations will remove small gradient (slope) [37, 112, 113].

Band Reject (BR) attempts to filter out only the trouble sum range of frequencies associated with noise. The notch to do this adds other low freq distortions. Texture predominately exists in intensity not as much between colors. This is why the band reject does
not alter the colors in LUV space as much as it does for RGB space. FFT also distorts color due to out of range results and forced normalization.

Iterative steering Kernel [130]

Savitzky-Golay (GS) [131, 132]


Figure J.1.37: Comparison of texture removal operations. Color space left to right: Gray, RGB, Luv.


Figure J.1.38: Comparison of texture removal operations. Color space left to right: Gray, RGB, Luv.


Figure J.1.39: Comparison of noise removal operations where $\sigma$ represents the standard deviation of the difference between the original and reconstructed blocks. Methods $\mathrm{d} / \mathrm{g} / \mathrm{j}$ remove the noise while preserving the ridge and edges. Methods $\mathrm{e} / \mathrm{h} / \mathrm{k}$ remove the noise and ridge while preserving the edges. Methods $\mathrm{c} / \mathrm{f} / \mathrm{i} / / \mathrm{m}$ smooth everything though each prefers different frequency ranges.

## J.1.10 Spectral Analysis

[87, 133]

Affinity Matrix [134]

## Spectrum

## J.1.11 Probability of Boundary

Cue Combination in classifier [68, 120, 135]

Multiscale Probability of Boundary (mPb) linear combination of brightness, color, and texture gradients at different radii $[31,41,68,134,136,137]$

- Get Energy @ each radius @ each ori
- Smooth Filter @ each radius @ each ori
- Combine weight @ each ori
- Combine ori track theta @ max energy
- NonMax Suppression

Spectral Probability of Boundary (sPb) Eiganvectors? and Oriented Energies on mPb [31, 41, 134, 136, 137]

Globalized Probability of Boundary (gPb) [31,41, 134, 136, 137]

## J. 2 Segmentation

Segmentation is the process of breaking an image apart into its important constituent visual components. Each segmentation algorithm uses particular feature set(s) to determine the optimal segmentation. A feature set is generally a low level way of quantifying
aspects of an image for later processing. Basic features of an image include color space, texture, and gradient. Each segmentation algorithm works like an optimization algorithm using the feature set to find the best fit or segmented regions.

Throughout this document the images shown in Figure J.2.1 are used because they contain many of the important features and other universally problematic characteristics. Dithering is often used in the creation of images to enhance their natural look by generating noise around boundaries. Many edge detection and segmentation algorithms get confused when boundaries are obscured with dithering or smoothing. The algorithms also get confused by the reflection of an object or when a single object of interest has multiple features such as color or texture.

(c) Dithering, Slow gradient, Same object multi-colored

Figure J.2.1: Images showing major challenges in image segmentation.

## J.2.1 Edge Detection

Edge detection algorithms attempt to find areas of rapid change in color (edges). This can simply be a threshold on any energy magnitude or boundary probability or use other more complicated calculations. [89,98, 138, 139]

Basic Just threshold an energy magnitude or boundary probability. This is fast but results in thick edges and is often susceptible to noise.

Canny fast, thin edges, hysteresis threshold, noise suppression

- Filter (Smooth)
- Gradient
- Non-maximum suppression
- Hysteresis thresholding
[88-91]

Marr-Hildreth Thin edges, susceptible to noise

$$
\begin{equation*}
E^{*}=Z \operatorname{eroCross}\left\{L o G * I_{x, y}\right\} \tag{J.2.1}
\end{equation*}
$$

Shen-Castan Thin edges, hysteresis threshold, susceptible to noise

- Infinite Symmetric Exponential Filter
- False zero-cross suppression
- Hysteresis thresholding
- Handles poorer SNR

$$
\begin{equation*}
f=a e^{-p(|x|+|y|)} \tag{J.2.2}
\end{equation*}
$$

[90]

SUSAN $[112,114,139,140]$ noise suppression, thick edges

Phase Congruency noise suppression, thick edges
[111]

$$
\begin{equation*}
M=\frac{1}{2}\left(c+a+\sqrt{b^{2}+(a-c)^{2}}\right) \tag{J.2.3}
\end{equation*}
$$

Edge Detection Comparison In figures where thresholding was used: the optimal threshold was found for RGB then retained for gray and Luv.


Figure J.2.2: Comparison of edge detectors. Color space left to right: Gray, RGB, Luv.


Figure J.2.3: Comparison of edge detectors. Color space left to right: Gray, RGB, Luv.

## J.2.2 Corner Detect

Corner detection is the process of finding straight edges that come together at some angle [89, 98, 138, 139].

Minimum Eigen Value $[67,111,141]$ "The block calculates the smaller eigenvalue of the sum of the squared difference matrix. This minimum eigenvalue corresponds to the corner metric matrix." [67]

$$
\begin{align*}
& A=\left(I_{x}\right)^{2} * w \\
& B=\left(I_{y}\right)^{2} * w  \tag{J.2.4}\\
& C=\left(I_{x} I_{y}\right) * w \\
& M=\begin{array}{ll}
A & C \\
C & B
\end{array} \tag{J.2.5}
\end{align*}
$$

Harris Harris (speed) [67, 111, 141]

$$
\begin{align*}
R=\operatorname{det}(M)-k(\operatorname{tr}(M))^{2} & =A B-C^{2}-k(A+B)^{2} \\
\operatorname{det}(M) & =\lambda_{1} \lambda_{2}  \tag{J.2.6}\\
\operatorname{trace}(M) & =\lambda_{1}+\lambda_{2}
\end{align*}
$$

Phase Congruency Phase Concurrency [111]

$$
\begin{equation*}
m=\frac{1}{2}\left(c+a-\sqrt{b^{2}+(a-c)^{2}}\right) \tag{J.2.7}
\end{equation*}
$$

SUSAN SUSAN (slow) [112, 114, 140]


Figure J.2.4: Comparison of corner detectors.


Figure J.2.5: Comparison of corner detectors.


Figure J.2.6: Comparison of corner detectors.

## Corner Detect Comparison

## J.2.3 Clustering Methods

These methods are based on grouping like pixels together in a given continuous feature space such as color and/or texture.

## Histogram

## Hierarchical Clustering [142, 143]

Spectral Clustering [133,144,145]

Mean Shift This algorithm assigns a pixel to a group by calculating density gradients. Each pixel assignment is iteratively moved closer to the nearest density sink [32,146,147]

Table J.2.1: Pros and cons of Mean Shift.

| Pros |  | Cons |
| :--- | :--- | :--- |
|  |  | Resource intensive |
|  | Oversegmentation |  |
|  | Gradient based features |  |
|  | Iterative |  |



Figure J.2.7: Mean Shift clusering on image of Roman Aqueducts ( $82 \mathrm{~min}, 37$ clusters)

Density-based spatial clustering of applications with noise (DBSCAN) Like Mean-
Shift this algorithm is based on density and is computationally expensive but it also accounts for continuity in the clustering space. [142, 148-151]

K-Means Not always same results due to random init, though matlab can use results from a small subset to initialize the full pixel set. [33,37,44]

$$
\begin{equation*}
p_{i} \in C_{k} \text { where } k=\underset{1 \leq j \leq n}{\operatorname{Arg} \min }\left\{\left\|p_{i}-\mu_{j}\right\|\right\} \tag{J.2.8}
\end{equation*}
$$

Table J.2.2: Pros and cons of K-Means.

| Pros |  | Cons |
| :--- | :--- | :--- |
|  |  | Gradient based features only |
|  | Random initialization |  |
|  | Inconsistant results |  |
|  | Iterative |  |
|  |  |  |

## Spacialy Constrained K-Means $[37,56]$

Dirichlet Process Gaussian Mixture Model (DPGMM) Classifier [152]


Figure J.2.8: Comparison of GMM and DPGMM [152]

## Expectation Maximization (EM) Classifier [153, 154]

Adaptive Clustering [155]

## Adding Spatial Components [87]



Figure J.2.9: Comparison of standard clustering algorithms in their clustering space [142]


Figure J.2.10: Comparison of clustering algorithms applied to image of Roman aqueduct. Color space left to right: Gray, RGB, Luv.


Figure J.2.11: Comparison of clustering algorithms applied to image of hot air balloons. Color space left to right: Gray, RGB, Luv.

## Cluster Comparison

Splitting Clusters Splitting clusters [44] Though initially used with k-mean this method can be used with other segmentation algorithms. It iteratively splits highest variance cluster and reruns k-mean until highest validity reached.

$$
\begin{gather*}
d_{\text {intra }}=\frac{1}{N} \sum_{j} \sum_{p_{i} \in C_{j}}\left\|p_{i}-\mu_{j}\right\|^{2}  \tag{J.2.9}\\
d_{\text {inter }}=\min _{i, j}\left\|\mu_{i}-\mu_{j}\right\|^{2}  \tag{J.2.10}\\
\text { Validity }=\frac{d_{\text {intra }}}{d_{\text {inter }}} \tag{J.2.11}
\end{gather*}
$$

## Combining Clusters [156]

Perceptual Boarder Refinement [38] Though initially used with k-mean this method can be used with other segmentation algorithms.

Effect of Initialization [157]

## J.2.4 Graphing Methods

These methods treat the pixels like nodes in a graph with edges connecting them. The methods then use different criteria to determine which edges to break apart until regions become secluded from the rest. [158]

Low contrast strong coupling

High contrast weak coupling


Figure J.2.12: Example of graphing method [158]

Min-Cuts [34]

Normalized Cuts NP-hard [35, 159, 160]

SE Min-Cuts $[34,40]$

Felzenszwalb \& Huttenlocher [23] Over segmentation

Max-flow \& Potts model Network model in place of graph [160-162]

## J.2.5 Contour Detection

Contour Completion [163,164]

Watershed Transformation $\quad[28,31,41,134,136,137]$

Ultrametric Contour Map [28,31,41,134,136,137]

## J.2.6 Differential Equations

Region Growing Now built into Matlab, only foreground/background based on user initial input [165]

Active Contours Now built into Matlab, only foreground/background based on user initial input. Using the Euler-Lagrange equation the integral equation can be solved as a differential equation. [48, 166]

$$
\begin{equation*}
E_{\text {snake }}(C)=\underset{c}{\operatorname{Arg} \min } \int_{0}^{1} \alpha(s)\left|C_{s}\right|^{2}+\int_{0}^{1} \beta(s)\left|C_{s s}\right|^{2}-\gamma \int_{0}^{1}|\delta I(C(s))| \tag{J.2.12}
\end{equation*}
$$

$$
\begin{gather*}
y_{o p t}=\underset{y}{\operatorname{Arg} \min } \int_{x_{1}}^{x_{2}} F\left(x, y, y^{\prime}, y^{\prime \prime}\right) d x  \tag{J.2.13}\\
\frac{\partial F}{\partial y}-\frac{d}{d x}\left(\frac{\partial F}{\partial y^{\prime}}\right)+\frac{d^{2}}{d x^{2}}\left(\frac{\partial F}{\partial y^{\prime \prime}}\right)
\end{gather*}
$$



Figure J.2.13: Example of active contour handling a subjective boundary [166]

Level Set $\quad 2^{n}$ Segments, Many input parameters [36, 43, 46, 48, 167-175]

$$
\begin{gather*}
\Psi(x, y)= \pm d(x, y) \\
\Psi_{t}+F(x(t))|\nabla \Psi|=0 \\
d \Psi=-F(x(t))|\nabla \Psi| d t \\
F(x(t) ; t)=\alpha\left(P_{f}(I)-P_{b}(I)\right)+(1-\alpha)(c+\kappa(x(t) ; t)) g(\nabla I)  \tag{J.2.14}\\
\kappa(x(t) ; t)=\operatorname{div}\left(\frac{\nabla \Psi}{|\nabla \Psi|}\right) \\
\frac{\partial \Psi}{\partial t}=\delta_{\epsilon}(\Psi)\left(\mu \kappa-\lambda_{1}\left(I-\mu_{1}\right)^{2}+\lambda_{2}\left(I-\mu_{2}\right)^{2}\right)
\end{gather*}
$$



Figure J.2.14: Example of level set and its ability to split/merge contours while being embed in 1 higher dimension [45]. Double click to open animation.

## J.2.7 Combined Methods

Global Probability of Boundary - Oriented Watershed Transform - Ultrametric Contour Map (gPb-owt-ucm) [31,41, 87, 136]

This method first finds the Multi-scale Probability of Boundary (mPb). This is done with a weighted sum of oriented gradients of histograms at different sizes performed on the
different color and texture dimensions. Next the Spectral Probability of Boundary (sPb) is found. This is done by determining eigen values/vectors associated with the image. Then the Global Probability of Boundary ( gPb ) is determined as a weighted sum of mPb and sPb . The gPb is then passed through an oriented watershed transform to convert boundaries to complete contours. Next the data is grouped in a hierarchical fashion with Ultrametric Contour Map.

Code requires linux and $5+G B$ of ram for single image, on the face only 1 parameter for the contour map however there are many lower parameters that were pre-determined based on their image set and are no longer accessible.

K-mean, Watershed, Difference in Strength [176]

Object-level [160]

## J. 3 Simplification

## J.3.1 General

Reduce Data [6,177-181]

Enhance/add data [182]

## J.3.2 Machine Learning

Generalized Linear Regression Model (GLM) [183]

Two Stage [184]

Logistic Regression [185, 186]

Ridge regularization and feature selection by adding bias to reduce variance and prevent over fitting and uses the L2 norm [64, 187, 188]

$$
\begin{equation*}
\min _{\beta} \sum_{i=1}^{n}\left(y_{i}-\sum_{j=1}^{p} \beta_{j} x_{i j}\right)^{2}+\lambda_{2} \sum_{j=1}^{p} \beta_{j}^{2} \tag{J.3.1}
\end{equation*}
$$

Least Absolute Shrinkage and Selection Operator (LASSO) bias like Ridge but also reduces parameters by eliminating variables with high correlation by using the L1 norm. This method handles sparse data. [64,187-189]


Figure J.3.1: LASSO (L1) vs Ridge (L2) [190]

$$
\begin{equation*}
\min _{\beta} \sum_{i=1}^{n}\left(y_{i}-\sum_{j=1}^{p} \beta_{j} x_{i j}\right)^{2}+\lambda_{1} \sum_{j=1}^{p}\left\|\beta_{j}\right\| \tag{J.3.2}
\end{equation*}
$$

Elastic Net balance between Ridge and LASSO $[64,188,191]$

$$
\begin{equation*}
\min _{\beta} \sum_{i=1}^{n}\left(y_{i}-\sum_{j=1}^{p} \beta_{j} x_{i j}\right)^{2}+\lambda_{1} \sum_{j=1}^{p}\left\|\beta_{j}\right\|+\lambda_{2} \sum_{j=1}^{p} \beta_{j}^{2} \tag{J.3.3}
\end{equation*}
$$

Principal Component Regression (PCR) feature selection by PCA used to reduce parameters by selecting the more influential based on covariance [64, 192, 193]

Partial Least Squares (PLS) feature selection by mapping input and output variables to a new space to best account for covariance [64, 192, 194]. PLS works on its own as a regression model just like LASSO but it also works like PCA to account for variances. PLS not only finds the cues that explain most of the input variance but also the output variance. PLS was non recoverable back to normal space to reduce cues. It would require the end version to use all cues but for speed in future user usage would want to use only the fewest.

Neural Net Non parametric, computationally expensive, highly susceptible to over fitting especially with smaller data set, not good with sparse data. After completion this article claims to have overcome these limitations which may be valuable in the future with continued work [195].

Fast Correlation Based Filter (FCBF) feature selection, though it does not handle biased/sparse data well yet [196].

Standardize vs Normalize accounts for variance and mean being drastically different cue metric scales. Normalization specs everything from 0 to 1 even if there are large outliers which compacts non-outlier data. Standardization scales values to fit a bell curve where some outliers could have very large or very small vales. [197]

$$
\begin{gather*}
X_{\text {Normalized }}=\frac{X-X_{\min }}{X_{\max }-X_{\min }} \\
X_{\text {Standardized }}=\frac{X-\mu}{\sigma} \tag{J.3.4}
\end{gather*}
$$

Leverage is when one or a few outlier observations cause drastic shifts in the predicted model and likely contribute to over fitting [67]

Cook's Distance Used to find outliers, represents how bad they fit the model and the group. Large values mean the observation is having a large effect on the model pulling it in a different direction from the condenses. "It summarizes how much all the values in the regression model change when the i observation is removed" [66,67]. $p$ is the number of cues in the model.

Residuals \& Fit are the errors left between the predicted model and the actual values. Patterns in residuals or fit can indicate missing feature cues. [67]

$$
\begin{equation*}
D_{i}=\frac{\sum_{j=1}^{n}\left(\hat{Y}_{j}-\hat{Y}_{j(i)}\right)^{2}}{p M S E} \tag{J.3.5}
\end{equation*}
$$

Cross-Validation Used to train on subset and test on remaining in an iterative manor with different subsets each time.

## J. 4 Assessment

## J.4.1 User Testing

Non-Inferior [198]

Odds vs Risk ratios [199, 200]

GEE Using SPSS Generalized Estimating Equations to calculate mean, significance, odds ratio, etc. [201]

ICC Intraclass Correlation [202]

NASA TLX [203]

SUS Systems Usability Survey [63,204]

Side by Side Ranking [205]

## J.4.2 Segmentation

Pascal challenge set forth some basic requirements [52].
Required Algorithm Properties for this research

- Computationally cheap
- Fast
- Minimal hardware/software resources
- Consistency - Each time same algorithm is run similar results are given
- Boundary Accuracy
- Detect Boundaries
- Minimal localization error with some tolerance
- Segment/Region Accuracy
- Detect Segments/Regions
- Minimal localization error with some tolerance
- Automatic
- Minimal human interaction required
- Set parameters that work for most images

Required Validation Properties

- Consistency
- Resulting metric is comparable across multiple images
- Resulting metric is comparable across multiple image sizes
- Accurately tests algorithm's required properties
- Help find the best parameter set for most images


## J.4.2.1 Boundaries

Hausdorff Distance is the distance between calculated and ground truth edge/boundary points [48,206-208]

$$
\begin{gather*}
B D E=d\left(p_{x, i}, Y\right)=\min _{j}\left\|p_{y, j}-p_{x, i}\right\|  \tag{J.4.1}\\
H D(S, G)=\max _{i} d\left(p_{s, i}, G\right)  \tag{J.4.2}\\
H D_{S y m}(S, G)=\max \left(\max _{i} d\left(p_{s, i}, G\right), \max _{i} d\left(p_{g, i}, S\right)\right) \tag{J.4.3}
\end{gather*}
$$

Loomis Distance is a description of difference between images that have been filtered to represent the sense of touch over boundaries [50].

$$
\begin{gather*}
c=0.163 \\
x, y=\text { square kernel } \\
\theta=(x c)^{2}+(y c)^{2}  \tag{J.4.4}\\
b=e^{-\pi \theta} \\
I_{b l u r}=I * b
\end{gather*}
$$

$$
\begin{equation*}
D=\left(\operatorname{sum}\left(\left(I_{b l u r, 1}-I_{b l u r, 2}\right)^{2}\right)\right)^{0.5} \tag{J.4.5}
\end{equation*}
$$

## J.4.2.2 Regions

These metrics are a measure of region overlap between calculated and ground truth regions.

Tanimoto coefficient (AKA: Jaccard index) [48]

$$
\begin{equation*}
T C\left(x_{i}, y_{j}\right)=\frac{\left|x_{i} \cap y_{j}\right|}{\left|x_{i} \cup y_{j}\right|} \tag{J.4.6}
\end{equation*}
$$

Dice Similarity Index [48]

$$
\begin{equation*}
\operatorname{DSI}\left(x_{i}, y_{j}\right)=\frac{2\left|x_{i} \cap y_{j}\right|}{\left|x_{i}\right|+\left|y_{j}\right|}=\frac{2 T C}{T C+1} \tag{J.4.7}
\end{equation*}
$$

Best Spatial Support [209]

$$
\begin{equation*}
B S S(S, G)=\frac{1}{N} \sum_{j}\left(\max _{i}\left(T C\left(s_{i}, g_{j}\right)\right)\right) \tag{J.4.8}
\end{equation*}
$$

Bidirectional Consistency Index [54, 207,210,211]

$$
\begin{gather*}
\operatorname{LRE}\left(X, Y, p_{i}\right)=\frac{\left|R\left(X, p_{i}\right) \backslash R\left(Y, p_{i}\right)\right|}{\left|R\left(X, p_{i}\right)\right|}  \tag{J.4.9}\\
L C E(S, G)=\frac{1}{N} \sum_{i} \min \left(L R E\left(S, G, p_{i}\right), L R E\left(G, S, p_{i}\right)\right)  \tag{J.4.10}\\
G C E(S, G)=\frac{1}{N} \min \left(\sum_{i} \operatorname{LRE}\left(S, G, p_{i}\right), \sum_{i} \operatorname{LRE}\left(G, S, p_{i}\right)\right) \tag{J.4.11}
\end{gather*}
$$

$$
\begin{equation*}
B C E(S,\{G\})=\frac{1}{N} \sum_{i} \min _{k}\left(\max \left(L R E\left(S, G_{k}, p_{i}\right), L R E\left(G_{k}, S, p_{i}\right)\right)\right) \tag{J.4.12}
\end{equation*}
$$

$$
\begin{equation*}
B C I=1-B C E \tag{J.4.13}
\end{equation*}
$$

## J.4.2.3 Information

These metrics are based on determining how much information calculated and ground truth regions have in common.

Square Difference Only good when labels have the same meaning between images. [48]

$$
\begin{equation*}
M S D(X, Y)=\frac{1}{N} \sum_{i}\left(p_{x, i}-p_{y, i}\right)^{2} \tag{J.4.14}
\end{equation*}
$$

Cross Correlation Only good when labels have the same meaning between images but allows for a shift in numbers. [48]

$$
\begin{equation*}
N C C(X, Y)=\frac{\sum_{i}\left(p_{x, i}-\mu_{1}\right) \sum_{i}\left(p_{y, i}-\mu_{2}\right)}{\left(\sum_{i}\left(p_{x, i}-\mu_{1}\right)^{2} \sum_{i}\left(p_{y, i}-\mu_{2}\right)^{2}\right)^{1 / 2}} \tag{J.4.15}
\end{equation*}
$$

Joint Entropy Used for other metrics. [48, 212]

$$
\begin{gather*}
H(X)=-\sum_{i}\left(P\left(x_{i}\right) \log \left(P\left(x_{i}\right)\right)\right)  \tag{J.4.16}\\
H(X, Y)=-\sum_{i} \sum_{j}\left(P\left(x_{i}, y_{j}\right) \log \left(P\left(x_{i}, y_{j}\right)\right)\right) \tag{J.4.17}
\end{gather*}
$$

$$
\begin{equation*}
H(X \mid Y)=-\sum_{i} \sum_{j}\left(P\left(x_{i}, y_{j}\right) \log \left(\frac{P\left(x_{i}, y_{j}\right)}{P\left(y_{j}\right)}\right)\right) \tag{J.4.18}
\end{equation*}
$$

Mutual Information Can determine correlation between dissimilar number schemes, modalities, or labeling. [48,212, 213]. Other version exist to improve response to large number of clusters: V-measure [214] which normalizes by sum of labeled inventories and adjusted version [142] that normalizes by chance using an expected score based on several factors to do the normalization.

$$
\begin{gather*}
M I(S, G)=H(S)+H(G)-H(S, G)  \tag{J.4.19}\\
N M I(S, G)=\frac{M I(S, G)}{H(S, G)} \tag{J.4.20}
\end{gather*}
$$

Variation of Information Effectively inverse of MI. [212,213,215-217]

$$
\begin{gather*}
V I(S, G)=H(S)+H(G)-2 M I(S, G)  \tag{J.4.21}\\
N V I(S, G)=1-\frac{M I(S, G)}{H(S, G)} \tag{J.4.22}
\end{gather*}
$$

Probabilistic Rand Index Good at penalizing for too many small regions. [210-212, 218-221] Adjusted version [216] also exists but requires an expected score based on several factors to normalize against chance.
a 2 pixels in same region in $S$ and same region in $G$, b 2 pixels in different region in $S$ and different region in G.

$$
\begin{gather*}
R I(S, G)=\frac{a+b}{a+b+c+d}=\frac{a+b}{\binom{N}{2}} \\
a+b=\sum_{\substack{i, j \\
i \neq j}}\left(\mathbb{I}\left(\left(s_{i}=s_{j}\right) \wedge\left(g_{i}=g_{j}\right)\right)+\mathbb{I}\left(\left(s_{i} \neq s_{j}\right) \wedge\left(g_{i} \neq g_{j}\right)\right)\right)  \tag{J.4.23}\\
c+d=\sum_{\substack{i, j \\
i \neq j}}\left(\mathbb{I}\left(\left(s_{i}=s_{j}\right) \wedge\left(g_{i} \neq g_{j}\right)\right)+\mathbb{I}\left(\left(s_{i} \neq s_{j}\right) \wedge\left(g_{i}=g_{j}\right)\right)\right)
\end{gather*}
$$

Table J.4.1: S and G contingency table (confusion matrix).

| Label | $S_{1}$ | $S_{2}$ | $\cdots$ | $S_{j}$ | $\sum$ |
| :---: | :---: | :---: | :---: | :---: | :---: |
| $G_{1}$ | $n_{11}$ | $n_{12}$ | $\cdots$ | $n_{1 j}$ | $u_{1}$ |
| $G_{2}$ | $n_{21}$ | $n_{22}$ | $\cdots$ | $n_{2 j}$ | $u_{2}$ |
| $\vdots$ | $\vdots$ | $\vdots$ | $\ddots$ | $\vdots$ | $\vdots$ |
| $G_{i}$ | $n_{i 1}$ | $n_{i 2}$ | $\cdots$ | $n_{i j}$ | $u_{i}$ |
| $\sum$ | $v_{1}$ | $v_{2}$ | $\cdots$ | $v_{j}$ | $N$ |

$$
\begin{gather*}
a+b=\binom{N}{2}+\sum_{i} \sum_{j} n_{i j}^{2}-\frac{1}{2}\left(\sum_{i} u_{i}^{2}+\sum_{j} v_{j}^{2}\right) \\
c+d=S S D=1-a+b=\frac{1}{2}\left(\sum_{i} u_{i}^{2}+\sum_{j} v_{j}^{2}\right)-\sum_{i} \sum_{j} n_{i j}^{2}  \tag{J.4.24}\\
P R I(S,\{G\})=\frac{\hat{a}+\hat{b}}{\binom{N}{2}}=\frac{1}{K} \sum_{k} R I\left(S, G_{k}\right) \\
\hat{a}+\hat{b}=\sum_{\substack{i, j \\
i \neq j}} c_{i j} p_{i j}+\left(1-c_{i j}\right)\left(1-p_{i j}\right)  \tag{J.4.25}\\
c_{i j}=\mathbb{I}\left(s_{i}=s_{j}\right) \\
p_{i j}=P\left(g_{i}=g_{j}\right)=\frac{1}{K} \sum_{k} \mathbb{I}\left(g_{k, i}=g_{k, j}\right)
\end{gather*}
$$

$$
\begin{gather*}
N P R I(S, G)=\frac{P R I-\mathbb{E}(P R I)}{\mathbb{M}(P R I)-\mathbb{E}(P R I)} \\
\mathbb{E}(P R I)=\frac{1}{\binom{N}{2}} \sum_{\substack{i, j \\
i \neq j}}\left(p_{i j}^{\prime} p_{i j}+\left(1-p_{i j}^{\prime}\right)\left(1-p_{i j}\right)\right)  \tag{J.4.26}\\
p_{i j}^{\prime}=\mathbb{E}\left(\mathbb{I}\left(s_{i}=s_{j}\right)\right) \approx \frac{1}{\Phi} \sum_{\phi} \frac{1}{K} \sum_{k} \mathbb{I}\left(g_{\phi, k, i}=g_{\phi, k, j}\right) \\
\operatorname{LPR}(\{S\})=\frac{1}{K} \sum_{k} P R I\left(S_{k}, S_{\{\neq k\}}\right) \tag{J.4.27}
\end{gather*}
$$



Figure J.4.1: How scores are affected by only a change in the number of clusters/regions [142]

## J.4.2.4 Classification

The metrics are based on first determining whether or not a point is correctly label (generally edge vs background) then calculating the rate of correct vs incorrect labels.

Pixel Correspondence is the attempt to determine whether or not a boundary pixel matches the ground truth boundaries. This is generally restricted with a one-to-one relation and a maximum distance. Leaving off the one-to-one relation can in some applications over inflate the perceived correctness of a result. [158]


Figure J.4.2: Pixel Correspondence [158]

Receiver operating characteristic Abandoned in favor of PR because of SR ratio causing most algorithms to falsely have high accuracy. [68,223-226]



| TP | FP |
| :--- | :--- |
| FN | TN |
| 1 | 1 |

Figure J.4.3: ROC [227]

$$
\begin{gather*}
T P R=\text { Sensitivity }=\frac{T P}{T P+F N} \\
T N R=\text { Specificity }=\frac{T N}{F P+T N}  \tag{J.4.28}\\
A C C=\text { Accuracy }=\frac{T P+T N}{P+N}
\end{gather*}
$$

Matthews correlation coefficient (MCC) [228]

$$
\begin{gather*}
M C C=\sqrt{\frac{X^{2}}{n}} \\
M C C=\frac{T P * T N-F P * F N}{\sqrt{(T P+F P)(T P+F N)(T N+F P)(T N+F N)}} \tag{J.4.29}
\end{gather*}
$$

Precision-recall (PR) In general a better F score or a score closer to the human point (ground truth) is a better preforming algorithm. The human point is the internal consistency among human raters. Though F score can be used for more than binary classification the segment labels are arbitrary and measuring their match with ground truth would be meaningless. In this case F score is used to match the correct classification of boundaries between computer vs human segmentation. [68, 193,218,223,226,229-231]


Figure J.4.4: PR [232]

$$
\begin{gather*}
P P V=\text { Precision }=\frac{T P}{T P+F P} \\
T P R=\text { Recall }=\frac{T P}{T P+F N} \\
F_{1}=2 \frac{P R}{P+R}  \tag{J.4.30}\\
F=\frac{P R}{\tau R+(1-\tau) P} \\
G=\sqrt{P R}
\end{gather*}
$$



Figure J.4.5: PR curve [158]

ROC/PR for Images Curves are based on trying a range of parameters and calculating ROC/PR points then connecting them. This represents a cost trade off curve. When multiple parameters are varied the line can be found by keeping only the best parameters then keeping only the max radius within discrete angle groupings. This represents a trade off front from a trade off point cloud. [135] The difference between human points in

Figures J.4.6 and J.4.7 comes from the way authors decided to create the ground truth images, number of ground truth illustrators, and whether or not pixel correspondence is accounted for.


Figure J.4.6: ROC and PR points to curve


Figure J.4.7: Example PR curve [31]

Box plot Box plots can be used to show statistical information. They are used here to show statistical information about algorithm performance across a single parameter change.


Figure J.4.8: Meaning of each component of the boxplot [233]. Notches represent $95 \%$ confidence interval.


Figure J.4.9: Example data shown with boxplot

## J.4.3 Simplification

## Adjusted $\mathbf{R}^{2}$

## Appendix K: Related Works

This chapter outlines related work from VCU Haptics and Rehabilitation Technologies Laboratory and other work mentored by Tyler Ferro.

## K. 1 Publications/Workshops

Providing Dynamic Access to Electronic Tactile Diagrams Ferro, T., Pawluk, D. (2017), International Conference on Universal Access in Human-Computer Interaction, HCII 2017

A significant problem for individuals who are blind or visually impaired is the lack of access to graphical information. In this paper, we describe our work on components of a system to make this access available in real-time, on demand, and through effective means. We start by discussing our current work on con-verting visual diagrams and images into a representation that can be more effec-tively interpreted by individuals who are blind or visually impaired. We then de-scribe previous and ongoing work in our laboratory on computer I/O devices we are developing to provide the given representation to the user immediately and in-teractively. Finally, we describe dynamic methods that we have developed to help manage the information presented more effectively given the constraints of the tactile system. [234]

## Developing Tactile Diagrams with Electronic Drawing Programs Using a Validated

 Texture Palette Ferro, T., Pawluk, D. (2015), Becoming Agents of Change, AER Annual conferenceThis presentation introduces the basics of computer aided tactile graphics creation through Photoshop/GIMP while focusing on how to include textures in the final product. In an effort to standardize and streamline tactile graphics
many new resources have become available for computers. Such advancements have often helped teachers save time when producing tactile graphics but also tend to leave texture behind. This presentation will teach the creation of tactile graphics with basic textures using Photoshop/GIMP. [5]

Automatic Image Conversion to Tactile Graphic Ferro, T., Pawluk, D. (2013), Computers and Accessibility, ASSETS '13 Proceedings of the 15th International ACM SIGACCESS Conference.

Currently, individuals who are blind or visually impaired have limited resources to allow them to interpret information contained in images. The aim of this project is to provide an accessible system to automatically generate tactile graphics for those who need to interpret information contained in visual images. The fundamental steps to accomplish this are to segment and simplify the image. The focus of this paper will be on several methods to segment an image. [235]

## K. 2 Concurrent

Perspective Correction Galen Kellner, Tyler Ferro, Dianne Pawluk
Worked on removing perspective from images automatically by detecting vanishing points.

Effectiveness of Using Local Cues to Indicate Perspective in Tactile Diagrams Andrea Nguyen, Megan Dell, Ramya Nandigam, Jacqueline Gabr Chavez Orellana, Tyler Ferro, Dianne Pawluk

This study presents and evaluates the use of a method using local cues to indicate perspective in tactile diagrams as compared to the current use of visual perspective methods. Perspective for an object using local cues is rep-
resented with standard visual perspective lines but with the thickness of the lines varying as a function of depth away from the viewer. Performance of study participants who were blind or visually impaired using the new method and the standard visual perspective method were compared as a function of: blindness onset of a participant, perspective method used, repetition (blocks were repeated twice), object and perspective of an object presented. For the main task, the method used (Wald $\chi^{2}=7.147, p=0.008$ ) and the methodrepetition interaction (Wald $\chi^{2}=4.272, p=0.039$ ) had a significant effect. Participants performed better with our new method and their improvement for (only) this method showed a significant improvement between repetitions. The findings demonstrate that our new method improved the performance of users for tasks involving perspective on diagrams over the standard visual perspective method. The data also indicates that with more practice, improvement could become even greater than observed during this study. Perspective frequently plays a critical role in aiding the understanding of questions in mathematics and science. TVIs who make tactile diagrams that need to include perspective can use our new method to present the perspective more clearly than in current diagrams. [236]

Tactile Display of Perspective - In Practice lan linville, Tyler Ferro, Dianne Pawluk Considering and representing perspective, where viewers are given the impression of height, width and depth of an object from a particular viewing angle (e.g., Figure 1), often makes tactile diagrams difficult to decode. Perspective, however, frequently plays a critical role in aiding the understanding of questions in mathematics and science, and is often found on many standardized tests and assessments. The difficulty with using touch, as compared to vision, is that the field of view is much smaller, which makes interpretation of global information, like perspective, more difficult. We will describe a
method that was developed in our laboratory that creates local cues in the tactile diagram to provide information about perspective. We will also present results from a study we performed that evaluated our new method in comparison to using standard visual perspective methods for tactile diagrams for use by individuals who are visually impaired.

As a method of representation is only useful if it can be generated easily, we have created software tools to aid with the creation of perspective diagrams using our new method. The method uses SketchUp, a software package that is easy for creating 3D models. SketchUp is free to educators. To this, we have created an Add-in Module, which we provide for free, that will add the local cues for our method automatically. SketchUp can then be used to select a view point to produce a 2-D drawing of the model, complete with the added local cues. The second half of the presentation will walk participants through the installation of the software tools, as well as a few examples using the software to generate 2-D perspective drawings with the added local cues. [237]

## Tactile Crayons Suraj Kandalam, Tyler Ferro, Dianne Pawluk

This paper describes the development and initial assessment of a set of tactile crayons that can produce different textured lines and areas on regular paper. The development of the tactile crayons focused on being able to differentiable along the texture dimension of sticky-slippery. Needs assessment through the use of surveys and focus groups guided the development. The ability of study participants to identify the textures of the crayons on paper was used as an initial assessment. User needs assessment showed many practitioners felt that the development of tactile crayons would aid in teaching and fill a needs gap. Testing showed that six tactile crayons performed with a mean accuracy of $77 \%$ (SE 4\%) in user discrimination. Further test-
ing demonstrated that the four main tactile crayons performed with a mean accuracy of $86 \%$ (SE 3\%) in user identification. Many parents and teachers showed interest in the potential of tactile crayons as a learning tool. The four main tactile crayons were highly discriminable and identifiable by adults who are blind or visually impaired. This shows promise for discriminability and identifiability of these crayons by children who are blind and visually impaired, and their potential as a learning tool in both formal and informal learning environments. Tactile crayons have the potential to aid childhood development and student learning. Tactile crayons could reduce art and school supply costs. Using tactile crayons could help reduce time needed by teachers to produce tactile diagrams. [238]

Research Assistant Megan Lavery, Heather Youmans, Tyler Ferro, Dianne Pawluk Assisted with this project by implementing user testing protocols and preparing images for digital processing.

## K. 3 Past

Automatic Haptics Display and Dynamic Zooming Ravi Rastogi, Dianne Pawluk One possibility of providing access to visual graphics for those who are visually impaired is to present them tactually: unfortunately, details easily available to vision need to be magnified to be accessible through touch. For this, we propose an "intuitive" zooming algorithm to solve potential problems with directly applying visual zooming techniques to haptic displays that sense the current location of a user on a virtual diagram with a position sensor and, then, provide the appropriate local information either through force or tactile feedback. Our technique works by determining and then traversing the levels of an object tree hierarchy of a diagram. In this manner, the zoom steps adjust
to the content to be viewed, avoid clipping and do not zoom when no object is present. The algorithm was tested using a small, "mouse-like" display with tactile feedback on pictures representing houses in a community and boats on a lake. We asked the users to answer questions related to details in the pictures. Comparing our technique to linear and logarithmic step zooming, we found a significant increase in the correctness of the responses (odds ratios of 2.64:1 and 2.31:1, respectively) and usability (differences of $36 \%$ and $19 \%$, respectively) using our "intuitive" zooming technique. [239]

## Haptics Texture Display David Burch, Dianne Pawluk

We present a multi-finger 2-D haptic display device to potentially provide the opportunity of parallel processing of coarse haptic information, with the intended benefit of improving performance in 2-D haptic object recognition. Combined with a novel method for producing "texture" image representations, we found that multiple contacts does improve performance compared to a single contact in terms of both time and accuracy. However, we also found no benefit for additional contacts with only raised-line representations of objects. This suggests that the use of texture made a key contribution in improving performance from one to multiple fingers. [49]

## Appendix L: Image Set

Reasonable attempts made to secure open use images and images for public/academic use. Please contact original authors for any further publication/commercial use.

## L. 1 Training Images

Too many algorithm and professionally generated diagrams from images to show here ( $\sim 500,000$ ) only showing original images.









## L. 2 Testing Images

Too many algorithm and professionally generated diagrams from images to show here ( $\sim 250$ ) only showing original images.























## Appendix M: Code

Code freely available for academic use. For commercial use, contact original authors. Reasonable attempts to cite authors from adapted use code are shown above code they authored. Where authors are not listed, assume code is from this work. To extract code from this PDF in 7-Zip file form, double click link: Code.7z

## Appendix N: Acknowledgments

- VCU Biomedical Engineering Dept.
- Academic Advisory Committee
- Professional Advisory Committee (Janice Johnson, Kit Burnett)
- Undergraduate Research Assistants (Megan Lavery, Autumn Hughes, Will Gourley)
- NSF IIS Grant \#1218310
- IRB Review \#HM14522
- Columbia Lighthouse for the Blind
- VA Department for the Blind and Visually Impaired

Appendix O: Vita

## Tyler Ferro

## EDUCATION

| PhD | Virginia Commonwealth University |
| :--- | :--- |
| MBA | Virginia Commonwealth University |
| CERT | Virginia Commonwealth University |
| MS | University of Oxford |
| BS | Southern Illinois University Carbondale |
| BS | Southern Illinois University Carbondale |

MBA Virginia Commonwealth University
CERT Virginia Commonwealth University
MS University of Oxford
BS Southern Illinois University Carbondale

## RESEARCH

| Biomedical Engineering | $3.9 / 4$ | Aug 2012 | May 2018 |
| :--- | :--- | :--- | :--- |
| Information Systems | $3.8 / 4$ | Aug 2015 | May 2018 |
| Preparing Future Faculty |  | Aug 2012 | May 2018 |
| Biomedical Engineering |  | Sept 2010 | Sept 2011 |
| Electrical Engineering | 3.5/4 | Aug 2005 | May 2010 |
| Computer Engineering | 3.5/4 | Aug 2005 | May 2010 |

## VCU Haptics Lab - Department of Biomedical Engineering

Aug 2012 May 2018 Graduate Research Project - Design algorithm to convert visual image to tactile diagram Key skills - C/C++/C\#, Data Analysis, Machine Learning, Matlab, Mentorship, Python, R, SAS, SPSS

Governors Data Analytics Internship - VCU School of Business
Aug 2016 Dec 2016
Team Leader - Analyzed government data to discover novel insights to influence trajectory Key skills - Data Analysis, Frequency Analysis, Matlab, SAS, SPSS, SQL, Tableau, Trend Analysis

Oxford Centre for Human Brain Activity
May 2011 Sept 2011
Master's Research Project - Designed algorithm to remove eye related artifacts from MEG signal data Key skills - Data Analysis, ICA, Matlab, PCA,

## SIUC Psychology Department

Aug 2009 Dec 2009
Research Assistant - Studied learning and behavior of small animals
SIUC Electrical \& Computer Engineering Department
Aug 2008 Dec 2008
Research Assistant - Studied power consumption of various cache parameters on mobile processors

## EXPERIENCE

## VCU Innovation Gateway

Aug 2017 May 2018
Licensing Liaison - Transform university innovations into intellectual property and market opportunities
VCU School of Business - Department of Information Systems
April 2017 May 2018
Course Development Intern - Developing a course in data wrangling
Key skills - AWS, Business Intelligence, CART, Data Analysis, EN, GEE, GIS, GLM, ICA, R, @RISK, SAS, SPSS

## VCU Haptics Lab - Department of Biomedical Engineering

Aug 2014 May 2018
Research Assistant - Automatic image conversion to tactile graphic (NSF Grant)
Laboratory Manager - Mentor undergraduate research. Manage lab operations, scheduling, and technologies

## VCU School of Engineering <br> Jan 2015 May 2016

Undergraduate Research Mentor (DURI) - Instructed undergraduate in research methods
VCU School of Engineering
Aug 2012 Aug 2014
Teaching Assistant - Lectured, taught lab class, scored lab reports, and scored homework
Fuchs Lubricants Co. - Harvey, IL
Jan 2012 July 2012
Operations Temp - Implemented a preventative maintenance program, designed and implemented a bulk-storage tracking database, programed an automated dissection support shipping and production schedule, and upgraded and automated various other systems
Key Skills - Access, Data Analysis, Operations Management, SAP, Supply Chain, SQL, VBA
SIUC University Housing - IT Office
Jan 2007 May 2010
Technician - Assisted University Housing employees with computer-related questions
Website Technician - Designed administrative web portals and applications
Key Skills - Active Directory, AJAX, BASH, PHP, Ruby, Ruby on Rails, SQL, Version Control, XML

VA AER 2018
Developing 3D Tactile Perspective Cues with Modeling Software (Undergraduate mentorship)
American Foundation for the Blind
JVIB 2018
Effectiveness of Using Local Cues to Indicate Perspective in Tactile Diagrams (Undergraduate mentorship)
Human-Computer Interaction Conference
HCII 2017
Providing Dynamic Access to Electronic Tactile Diagrams
VA Association for Education and Rehabilitation
Developing Tactile Diagrams with Electronic Drawing Programs
Computers and Accessibility Conference
ASSETS 2013
Automatic Image Conversion to Tactile Graphic

## EXTRACURRICULAR

National Institute of Justice - Crime Forecasting Challenge Jan 2017 May 2017
Team Leader - Developed data analysis model for predicting crime in Portland OR
Key Skills - Data Analysis, Frequency Analysis, GIS, Matlab, SPSS, Tableau, Trend Analysis
VCU Engineering World Health
May 2013 Aug 2016
Graduate Advisor - Taught leadership skills to undergraduate board
IT officer - Managed website, assisted with technology

## Oxford Biomedical Engineering Society

Nov 2010 Sept 2011
IT officer - Managed website, assisted with technology

## Oxford Engineering World Health

Nov 2010 June 2011
Designed low cost pulse oximeter for developing countries

## HONORS

## Virginia Commonwealth University

Special Recognition for Teaching Excellence - VCU School of Engineering
Honors Society - Tau Beta Pi
Scholarships - Great Lakes National, Illinois Scottish Rite Fund, Order of the Eastern Star of Illinois

## Southern Illinois University Carbondale

Honors Societies - Golden Key, Sigma Alpha Lambda
Scholarships - Illinois State Scholarship, Omron Electronics Inc.

## SKILLS

## Programming

Access, Actionscript, Active Directory, AJAX, Android, Ant, Bash, C/C++, C\#, CSS, DOM, Eclipse, GIT, HTML, Java, JavaScript, LaTeX, Linux, MATLAB, Multithread, MySQL, NetBeans, Perl, PHP, Python, Ruby, Ruby on Rails, SAP, Shell Script, Simulink, SMS, SQL, SVN, VB/VBA/VBS, Verilog, VHDL, Visual Studio, VLSI, Xilinx, XML

## Statistics/Modeling

CART, EN, GEE, GIS, GLM, ICA, LASSO, Level Set, Machine Learning, PCA, PLS, R, @RISK, SAS, SPSS

## Data Science

Amazon Web Services, Business Intelligence, Data Visualization, Frequency Analysis, IBM Watson, Machine Learning, Tableau, Trend Analysis

## Appendix P: Bibliography

[1] L. Hasty, "Textbooks Information - Personal Communication," 2013.
[2] World Health Organization, "Visual impairment and blindness," 2014. [Online]. Available: http://www.who.int/mediacentre/factsheets/fs282/en/
[3] National Federation of the Blind (NFB), "Statistical Facts about Blindness in the United States," 2014. [Online]. Available: https://nfb.org/blindness-statistics
[4] P. K. Edman, Tactile Graphics. American Foundation for the Blind, 1992. [Online]. Available: http://books.google.com/books?id=C7vq4-Th71AC\{\&\}|pg= PP1\{\&\}ots=ppgfq0qGFR\{\&\}dq=TactileGraphics\{\&\}pg=PP1\{\#\}v=onepage\{\&\}q= TactileGraphics $\{\&\}$ f=false
[5] T. Ferro and D. Pawluk, "Developing Tactile Diagrams with Electronic Drawing Programs Using a Validated Texture Palette," in Becoming Agents of Change, AER Annual conference, 2015. [Online]. Available: http://wp.vcu.edu/dtpawluklab/ workshops/
[6] R. E. Ladner, B. Slabosky, A. Martin, A. Lacenski, S. Olsen, D. Groce, M. Y. Ivory, R. Rao, S. Burgstahler, D. Comden, S. Hahn, M. Renzelmann, S. Krisnandi, and M. Ramasamy, "Automating tactile graphics translation," Proceedings of the 7th international ACM SIGACCESS conference on Computers and accessibility - Assets '05, p. 150, 2005. [Online]. Available: http://dl.acm.org/citation.cfm?doid=1090785.1090814
[7] T. P. Way and K. E. Barner, "Automatic visual to tactile translation-Part I: Human factors, access methods, and image manipulation." IEEE transactions on rehabilitation engineering : a publication of the IEEE Engineering in Medicine and Biology Society, vol. 5, no. 1, pp. 81-94, mar 1997. [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/9086389
[8] S. E. Krufka, K. E. Barner, and T. C. Aysal, "Visual to tactile conversion of vector graphics." IEEE transactions on neural systems and rehabilitation engineering : a publication of the IEEE Engineering in Medicine and Biology Society, vol. 15, no. 2, pp. 310-21, jun 2007. [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/17601201
[9] T. P. Way and K. E. Barner, "Automatic visual to tactile translation-Part II: Evaluation of the TACTile Image Creation System." IEEE transactions on rehabilitation engineering : a publication of the IEEE Engineering in Medicine and Biology Society, vol. 5, no. 1, pp. 95-105, mar 1997. [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/9086390
[10] C. Jayant, M. Renzelmann, D. Wen, S. Krisnandi, R. Ladner, and D. Comden, "Automated Tactile Graphics Translation: In the Field," ASSETS'07, pp. 75-82, 2007.
[11] S. E. Hemandez and K. E. Barner, "Joint Region Merging Criteria for WatershedBased Image Segmentaion," IEEE, 2000.
[12] K. Kokjer, "The Information Capacity of the Human Fingertip," IEEE Transactions on Systems, Man, and Cybernetics, vol. 17, no. 1, pp. 100-102, 1987.
[13] R. Manduchi and S. Kurniawan, Assistive Technology for Blindness and Low Vision, R. Manduchi and S. Kurniawan, Eds. CRC Press, 2012.
[14] H. T. C. T. Unit, "Creating Tactile Graphics," pp. 1-25.
[15] T. T. G. Project, "Tactile Graphics Assistant Software Reference \& Usage Manual," 2013.
[16] D. Center, Accessible Image Sample Book. Global Literacy Initiative, 2014.
[17] A. Slowinksi, "Tactile Graphic Image Library + Graphic Freeware = Meaningful Cost-Efficient Graphics." [Online]. Available: http://www.aph.org/training/ aslowinski/
[18] L. Hasty, "tactilegraphics.org." [Online]. Available: http://www.tactilegraphics.org/
[19] A. Turner, "Behind bars, Braille's dots fulfill prison inmates, aid the blind," dec 2012. [Online]. Available: http://www.houstonchronicle.com/news/houston-texas/ houston/article/Behind-bars-Braille-s-dots-fulfill-prison-4153070.php
[20] R. Ladner, "CSE 455: Automating Tactile Graphics Translation Computer Vision," 2010.
[21] ——, "CSE 455: Automating Tactile Graphics Translation," 2009.
[22] D. Lareau and J. Lang, "Instrument for haptic image exploration," IEEE Transactions on Instrumentation and Measurement, vol. 63, no. 1, pp. 35-45, 2014.
[23] P. F. Felzenszwalb and D. P. Huttenlocher, "Efficient Graph-Based Image Segmentation," International Journal of Computer Vision, vol. 59, no. 2, pp. 167-181, sep 2004. [Online]. Available: http://link.springer.com/10.1023/B: VISI.0000022288.19776.77
[24] Intel, "Color Models," in Inte ${ }^{\circledR}$ ® Integrated Performance Primitives for Intel $®$ ® Architecture Reference Manual, Volume 2: Image and Video Processing. Intel, 2009, ch. 6. [Online]. Available: http://software.intel.com/sites/products/ documentation/hpc/ipp/ippi/ippi\{_\}ch6/ch6\{_\}color\{_\}models.html
[25] J. Hartmanis and J. V. Leeuwen, Computer Vision âĂĬ ECCV 2002, ser. Lecture Notes in Computer Science, A. Heyden, G. Sparr, M. Nielsen, and P. Johansen, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, apr 2002, vol. 2352. [Online]. Available: http://link.springer.com/10.1007/3-540-47977-5
[26] L. G. Ugarriza, E. Saber, S. R. Vantaram, V. Amuso, M. Shaw, and R. Bhaskar, "Automatic image segmentation by dynamic region growth and multiresolution merging." IEEE transactions on image processing : a publication of the IEEE Signal Processing Society, vol. 18, no. 10, pp. 2275-88, oct 2009. [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/19535323
[27] R. C. Gonzalez, R. E. Woods, and S. L. Eddins, Digital Image Processing Using Matlab, 2nd ed. Gatesmark, LLC, 2009. [Online]. Available: http://books.google.com/books/about/ Digital\{_\}Image\{_\}Processing\{_\}Using\{\}Matlab.html?id=YYuJQgAACAAJ
[28] J. Malik, S. Belongie, J. Shi, and T. Leung, "Textons, contours and regions: cue integration in image segmentation," Proceedings of the Seventh IEEE International Conference on Computer Vision, vol. 2, no. September, 1999.
[29] K. Kulhavy, "Lisa analysis," 2012. [Online]. Available: http://upload.wikimedia.org/ wikipedia/commons/c/cd/Lisa\{_\}analysis.png
[30] A. Majumder, "Lateral Geniculate Nucleus (LGN)," pp. 1-33, 2010. [Online]. Available: http://www.ics.uci.edu/\{~\}majumder/vispercep/chap3\{_\}LGN\{_\}highvision. pdf
[31] P. Arbeláez, M. Maire, C. Fowlkes, and J. Malik, "From Contours to Regions: An Empirical Evaluation," in 2009 IEEE Conference on Computer Vision and Pattern Recognition. IEEE, jun 2009, pp. 2294-2301. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs\{_\}all.jsp?arnumber=5206707http:// ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5206707
[32] D. Comaniciu and P. Meer, "Mean Shift: A Robust Approach Toward Feature Space Analysis," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 24, no. 5, pp. 603-619, may 2002. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1000236
[33] C. Ding and X. He, "K-means Clustering via Principal Component Analysis," Twenty-first international conference on Machine learning - ICML '04, p. 29, 2004. [Online]. Available: http://portal.acm.org/citation.cfm?doid=1015330.1015408
[34] F. Estrada, A. Jepson, and C. Chennubhotla, "Spectral Embedding and Min-Cut for Image Segmentation," Procedings of the British Machine Vision Conference 2004, pp. 34.1-34.10, 2004. [Online]. Available: http://www.bmva.org/bmvc/2004/ papers/paper\{_\}119.html
[35] J. Malik and J. Shi, "Normalized cuts and image segmentation," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 22, no. 8, pp. 888-905, 2000. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm? arnumber=868688
[36] L. A. Vese and T. F. Chan, "A multiphase level set framework for image segmentation using the Mumford and Shah model," International Journal of Computer Vision, vol. 50, no. 3, pp. 271-293, 2002. [Online]. Available: http://www.springerlink.com/index/nt4622460602wqp1.pdf
[37] M. Luo, Y.-F. Ma, and H.-J. Zhang, "A spatial constrained k-means approach to image segmentation," in Information, Communications and Signal Processing, 2003 and Fourth Pacific Rim Conference on Multimedia. Proceedings of the 2003 Joint Conference of the Fourth International Conference on, vol. 2, 2003, pp. 738-742. [Online]. Available: http: //ieeexplore.ieee.org/xpls/abs\{_\}all.jsp?arnumber=1292554
[38] T. Pappas, A. Mojsilovic, and B. Rogowitz, "Adaptive perceptual color-texture image segmentation," IEEE Transactions on Image Processing, vol. 14, no. 10, pp. 1524-1536, oct 2005. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs\{_\}all.jsp?arnumber=1510687http: //ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1510687
[39] F. Estrada and A. Jepson, "Quantitative Evaluation of a Novel Image Segmentation Algorithm," IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), vol. 2, pp. 1132-1139, 2005. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1467570
[40] F. Estrada, "SE Min Cuts - Personal Communication," 2013.
[41] P. Arbeláez, M. Maire, C. Fowlkes, and J. Malik, "Contour detection and hierarchical image segmentation." IEEE transactions on pattern analysis and machine intelligence, vol. 33, no. 5, pp. 898-916, may 2011. [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/20733228
[42] S. E. Hernandez and K. E. Barner, "Tactile imaging using watershed-based
image segmentation," Proceedings of the fourth international ACM conference on Assistive technologies - Assets '00, pp. 26-33, 2000. [Online]. Available: http://portal.acm.org/citation.cfm?doid=354324.354332
[43] T. Brox and J. Weickert, "Level Set Based Image Segmentation with Multiple Regions," in Pattern Recognition, C. Rasmussen, H. Bülthoff, B. Schölkopf, and M. Giese, Eds., 2004, pp. 415-423.
[44] S. Ray and R. Turi, "Determination of number of clusters in k-means clustering and application in colour image segmentation," in Fourth International Conference on Advances in Pattern Recognition and Digital Techniques (ICAPRDT'99), 1999, pp. 137-143. [Online]. Available: http://www.csse.monash.edu.au/\{~\}roset/papers/ cal99.pdf
[45] C. Li, C. Xu, C. Gui, M. D. Fox, C. Li, C. Xu, C. Gui, and M. D. Fox, "Distance Regularized Level Set Evolution (DRLSE)," 2010. [Online]. Available: http://www.imagecomputing.org/\{~\}cmli/DRLSE/
[46] C. Li, C. Xu, C. Gui, and M. D. Fox, "Distance regularized level set evolution and its application to image segmentation," IEEE Transactions on Image Processing, vol. 19, no. 12, pp. 3243-3254, 2010.
[47] X. Xie and M. Mirmehdi, "Radial basis function based level set interpolation and evolution for deformable modelling," pp. 18-20, 2011. [Online]. Available: http://www.cs.swan.ac.uk/\{~\}csjason/rbf/
[48] A. Noble and J. A. Schnabel, "Univrsity of Oxford: BME Imageing Lectures," Oxford, 2010.
[49] D. Burch and D. Pawluk, "Using multiple contacts with texture-enhanced graphics," in 2011 IEEE World Haptics Conference. IEEE, jun 2011, pp. 287-292. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper. htm?arnumber=5945500
[50] J. M. Loomis, "A model of character recognition and legibility." Journal of experimental psychology. Human perception and performance, vol. 16, no. 1, pp. 106120, 1990.
[51] P. Arbeláez, M. Maire, C. Fowlkes, and J. Malik, "Contour Detection and Hierarchical Image Segmentation," Tpami, vol. 33, no. 5, pp. 898-916, 2011. [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/20733228
[52] M. Everingham and J. Winn, "The PASCAL Visual Object Classes Challenge 2012 ( VOC2012 ) Development Kit," pp. 1-32, 2012.
[53] F. J. Estrada and A. D. Jepson, "Benchmarking Image Segmentation Algorithms," International Journal of Computer Vision, vol. 85, no. 2, pp. 167-181, may 2009. [Online]. Available: http://www.springerlink.com/index/10. 1007/s11263-009-0251-z
[54] D. Martin, C. Fowlkes, D. Tal, and J. Malik, "A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics," in Proceedings Eighth IEEE International Conference on Computer Vision. ICCV 2001, vol. 2. IEEE Comput. Soc, 2001, pp. 416-423. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper. htm?arnumber=937655
[55] B. Hariharan and P. Arbelaez, "Semantic contours from inverse detectors," ...

Vision (ICCV), 2011 ..., 2011. [Online]. Available: http://ieeexplore.ieee.org/xpls/ abs\{_\}all.jsp?arnumber=6126343
[56] M. Mignotte, "A de-texturing and spatially constrained K-means approach for image segmentation," Pattern Recognition Letters, vol. 32, no. 2, pp. 359-367, jan 2011. [Online]. Available: http://linkinghub.elsevier.com/retrieve/pii/S016786551000320X
[57] T. Brox, M. Rousson, R. Deriche, and J. Weickert, "Unsupervised Segmentation Incorporating Colour, Texture, and Motion," Computer Analysis of Images and Patterns, no. August 2003, pp. 353-360, 2003. [Online]. Available: http://www.springerlink.com/content/3ghduabp4c5j3kc2
[58] L. I. Davitt, F. Cristino, A. C.-N. Wong, and E. C. Leek, "Shape information mediating basic- and subordinate-level object recognition revealed by analyses of eye movements." Journal of experimental psychology. Human perception and performance, vol. 40, no. 2, pp. 451-6, 2014. [Online]. Available: http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=3977674\{\&\}tool= pmcentrez\{\&\}rendertype=abstract
[59] G. W. Humphreys, C. J. Price, and M. J. Riddoch, "From objects to names: a cognitive neuroscience approach." Psychological research, vol. 62, no. 2-3, pp. 118-130, 1999.
[60] M. Mack, "The Dynamics of Categorization: Rapid Categorization Unraveled," Praca doktorska, vol. 144, no. August, pp. 551-569, 2011. [Online]. Available: http://etd.library.vanderbilt.edu/available/etd-05242011-101758/unrestricted/ Mack\{_\}dissertation\{_\}formatted.pdf
[61] G. Mather, F. Verstraten, S. Anstis, and D. J. Levitin, "This excerpt from The Motion Aftereffect . of MIT CogNet . Chapter 6 Experimental Design in Psychological Research," Psychological Research, pp. 115-130, 1998.
[62] E. Rosch and B. B. Lloyd, "Cognition and Categorization," Lloydia Cincinnati, vol. pp, pp. 27-48, 1978. [Online]. Available: http://www.worldcat.org/title/ cognition-and-categorization/oclc/3844382\{\&\}referer=brief\{_\}results
[63] Usability.gov, "System Usability Scale [SUS]." [Online]. Available: https: //www.usability.gov/how-to-and-tools/methods/system-usability-scale.html
[64] A. Acharjee, R. Finkers, R. G. Visser, and C. Maliepaard, "Comparison of Regularized Regression Methods for "Omics Data," Metabolomics, vol. 3, no. 3, p. 126, 2013. [Online]. Available: http://www.omicsonline.org/ comparison-of-regularized-regression-methods-for-omics-data-2153-0769-3-126. php $\{\&\}\{\&\}$ aid=20117
[65] A. Sima, "Feature Selection - Personal Communication," 2017.
[66] S. Glen, "Cook's Distance / Cook’s D: Definition, Interpretation," 2016. [Online]. Available: http://www.statisticshowto.com/cooks-distance/
[67] Mathworks, "Matlab Help Documentaion." [Online]. Available: http://www. mathworks.com/help/index.html
[68] D. R. Martin, C. C. Fowlkes, and J. Malik, "Learning to detect natural image boundaries using local brightness, color, and texture cues." IEEE transactions on pattern analysis and machine intelligence, vol. 26, no. 5, pp. 530-49, may 2004. [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/15460277
[69] NIH, "Competencies Proficiency Scale," 2009. [Online]. Available: https:
//hr.nih.gov/working-nih/competencies/competencies-proficiency-scale
[70] J. I. Siddique and K. E. Barner, "Nonlinear image decomposition for multiresolution edge detection using gray-level edge maps," Optical Engineering, vol. 41, no. 11, pp. 2749-2763, nov 2002. [Online]. Available: http: //opticalengineering.spiedigitallibrary.org/article.aspx?doi=10.1117/1.1512665
[71] Braille Authority of North America and Canadian Braille Authority, "Guidelines and Standards for Tactile Graphics , 2010," 2011. [Online]. Available: www.brailleauthority.org/tg
[72] J. Saldana, The Coding Manual for Qualitative Researchers. SAGE, 2009.
[73] M. Schreier, Qualitative Content Analysis in Practice. SAGE, 2012.
[74] D. T. V. Pawluk, R. J. Adams, S. Member, and R. Kitada, "Designing Haptic Assistive Technology for Individuals Who Are Blind or Visually Impaired," IEEE TRANSACTIONS ON HAPTICS, vol. 8, no. 3, pp. 258-278, 2015.
[75] a. Almansa, a. Desolneux, and S. Vamech, "Vanishing point detection without any a priori information," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 25, no. 4, pp. 502-507, 2003.
[76] V. Cantoni, L. Lombardi, M. Porta, and N. Sicard, "Vanishing point detection: representation analysis and new approaches," Proceedings 11th International Conference on Image Analysis and Processing, pp. 90-94, 2001.
[77] X. C. Yin, J. Sun, S. Naoi, K. Fujimoto, H. Takebe, Y. Fujii, and K. Kurokawa, "A multi-stage strategy to perspective rectification for mobile phone camera-based document images," Proceedings of the International Conference on Document Analysis and Recognition, ICDAR, vol. 2, pp. 574-578, 2007.
[78] S. C. Dakin and R. J. Watt, "The computation of orientation statistics from visual texture," Vision Research, vol. 37, no. 22, pp. 3181-3192, 1997.
[79] E. Ribeiro and E. R. Hancock, "Improved orientation estimation for texture planes using multiple vanishing points," Pattern Recognition, vol. 33, no. 10, pp. 15991610, 2000.
[80] M. Green, "SBFAQ Part 1: Basic Terms and Definitions," 2014. [Online]. Available: http://www.visualexpert.com/FAQ/Part1/cfaqPart1.html
[81] B. MacEvoy, "the geometry of color perception," 2009. [Online]. Available: http://www.handprint.com/HP/WCL/color2.html
[82] D. Briggs and R. Kristanto, "LIGHTNESS AND CHROMA," 2007. [Online]. Available: http://www.huevaluechroma.com/index.php
[83] D. Judd, "Hue, saturation, and lightness of surface colors with chromatic illumination," Journal of Research of the National Bureau of Standards, vol. 24, no. 3, p. 293, 1940.
[84] S. B. Staff, "Now You See It, Now You Don't: A Chromatic Adaptation Project," pp. 1-7, 2014. [Online]. Available: http://www.sciencebuddies.org/ science-fair-projects/project\{_\}ideas/HumBeh\{_\}p008.shtml
[85] R. Munroe, "Dress Color," 2015. [Online]. Available: https://xkcd.com/1492/
[86] R. Dony and S. Wesolkowski, "Edge detection on color images using RGB vector angles," Engineering Solutions for the Next Millennium. 1999 IEEE Canadian Conference on Electrical and Computer Engineering (Cat. No.99TH8411), vol. 2, pp. 687-692, 1999. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/
wrapper.htm?arnumber=808005
[87] M. R. Maire, "Contour Detection and Image Segmentation," Ph.D. dissertation, University of California, Berkeley, 2009. [Online]. Available: http://ttic.uchicago. edu/\{ $\sim$ \}mmaire/papers/pdf/mmaire\{_\}thesis.pdf
[88] J. Canny, "A Computational Approach to Edge Detection," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. PAMI-8, no. 6, pp. 679-698, nov 1986. [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/21869365http: //ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=4767851
[89] M. Nixon and A. Aguado, Feature Extraction \& image Proccessing, 2nd ed. Amsterdam: Academic Press, 2008.
[90] J. Parker, Algorithms for Image Proccessing and Computer Vision, 2nd ed., C. Long, Ed. Indianapolis, IN: Wiley, 2011.
[91] M. Basu, "Gaussian-based edge-detection methods-a survey," IEEE Transactions on Systems, Man and Cybernetics, Part C (Applications and Reviews), vol. 32, no. 3, pp. 252-260, aug 2002. [Online]. Available: http://ieeexplore.ieee.org/ Ipdocs/epic03/wrapper.htm?arnumber=1097737
[92] A. P. Burgess, "Towards a unified understanding of event-related changes in the EEG: the firefly model of synchronization through cross-frequency phase modulation." PloS one, vol. 7, no. 9, p. e45630, jan 2012. [Online]. Available: http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=3458099\{\&\}tool= pmcentrez\{\&\}rendertype=abstract
[93] D. Kim and H. Oh, "EMD: A package for empirical mode decomposition and hilbert spectrum," The R Journal, vol. 1, no. 1, pp. 40-46, 2009. [Online]. Available: http://journal.r-project.org/archive/2009-1/RJournal\{_\}2009-1\{_\}Kim+Oh.pdf
[94] L. Liang, J. Pu, and Z. Ping, "Texture segmentation using window empirical mode decomposition," in 2012 IEEE International Conference on Automation and Logistics, no. August. IEEE, aug 2012, pp. 373-377. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs\{_\}all.jsp?arnumber=6308238http: //ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6308238
[95] Z. Liu, H. Wung, and S. Peng, "Texture segmentation using directional empirical mode decomposition," in 2004 International Conference on Image Processing, 2004. ICIP '04., vol. 1. IEEE, 2004, pp. 279-282. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1418744
[96] A. Linderhed, "Variable Sampling of the Empirical Mode Decomposition of Two-Dimensional Signals," International Journal of Wavelets, Multiresolution and Information Processing, vol. 03, no. 03, pp. 435-452, sep 2005. [Online]. Available: http://www.worldscientific.com/doi/abs/10.1142/S0219691305000932
[97] J. Nunes, Y. Bouaoune, E. Delechelle, O. Niang, and P. Bunel, "Image analysis by bidimensional empirical mode decomposition," Image and Vision Computing, vol. 21, no. 12, pp. 1019-1026, nov 2003. [Online]. Available: http://linkinghub.elsevier.com/retrieve/pii/S0262885603000945
[98] K. R. Castleman, Digital Image Processing, T. Robbins, Ed. Upper Saddle River, New Jersey: Prentice Hall, 1996.
[99] J. M. S. Prewitt, "Object Enhancement and Extraction," in Picture Processing and Psychopictorics. Academic Press, 1970, pp. 75-146.
[100] R. Kirsch, "Computer determination of the constituent structure of biological images," Computers and biomedical research, vol. 328, pp. 315-328, 1971. [Online]. Available: http://www.sciencedirect.com/science/article/pii/0010480971900346
[101] D. Marr and E. Hildreth, "Theory of Edge Detection," Proceedings of the Royal Society B: Biological Sciences, vol. 207, no. 1167, pp. 187-217, feb 1980. [Online]. Available: http://rspb.royalsocietypublishing.org/cgi/doi/10.1098/rspb.1980.0020
[102] J. Malik, S. Belongie, T. Leung, and J. Shi, "Contour and texture analysis for image segmentation," International Journal of Computer Vision, vol. 43, no. 1, pp. 7-27, 2001. [Online]. Available: http://link.springer.com/article/10.1023/A: 1011174803800
[103] M. Landy and J. Bergen, "Texture segregation and orientation gradient," Vision research, vol. 31, no. 4, pp. 679-691, 1991. [Online]. Available: http://www.sciencedirect.com/science/article/pii/004269899190009T
[104] M. Unser, D. Sage, and D. Van De Ville, "Multiresolution monogenic signal analysis using the Riesz-Laplace wavelet transform." IEEE transactions on image processing : a publication of the IEEE Signal Processing Society, vol. 18, no. 11, pp. 2402-18, nov 2009. [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/19605325
[105] K. Rajpoot, V. Grau, and J. Noble, "Local-phase based 3D boundary detection using monogenic signal and its application to real-time 3-D echocardiography images," ... Imaging: From Nano to Macro, ..., pp. 783-786, 2009. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs\{_\}all.jsp?arnumber=5193166
[106] M. Felsberg and G. Sommer, "The Monogenic Scale-Space: A Unifying Approach to Phase-Based Image Processing in Scale-Space," Journal of Mathematical Imaging and Vision, vol. 21, no. 1, pp. 5-26, jul 2004. [Online]. Available: http://link.springer.com/10.1023/B:JMIV.0000026554.79537.35
[107] ——, "The monogenic signal," IEEE Transactions on Signal Processing, vol. 49, no. 12, pp. 3136-3144, 2001. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/ epic03/wrapper.htm?arnumber=969520
[108] S. Bernstein, J. Bouchot, M. Reinhardt, and B. Heise, "Generalized analytic signals in image processing: Comparison, theory and their applications," Quaternion and Clifford Fourier Transforms and Wavelets, vol. Pre-Publis, 2013. [Online]. Available: http://www.math.drexel.edu/\{~\}jb3455/publications/ICCA9GenAnalytic.pdf
[109] P. Kovesi, "Phase congruency: A low-level image invariant," Psychological Research, vol. 64, no. 2, pp. 136-148, dec 2000. [Online]. Available: http://www.ncbi. nlm.nih.gov/pubmed/11195306http://link.springer.com/10.1007/s004260000024
[110] ——, "Image features from phase congruency," Videre: Journal of computer vision research, no. June, 1999. [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/ download?doi=10.1.1.54.5658\{\&\}rep=rep1\{\&\}type=pdf
[111] -, "Phase congruency detects corners and edges," The australian pattern recognition society conference: ..., 2003. [Online]. Available: http://books.google.com/books?hl=en\{\&\}|r=\{\&\}id=oz-Sinxlj08C\{\&\}oi=fnd\{\&\}pg= PT381\{\&\}dq=Phase+Congruency+Detects+Corners+and+Edges\{\&\}ots= yXAwNEpUrV\{\&\}sig=uwhswuxEGINuTwjMcSMRAC8qQaw
[112] S. M. Smith and J. M. Brady, "SUSAN âĂŤ A New Approach to Low Level Image

Processing," International Journal of Computer Vision, vol. 23, no. 1, pp. 45-78, 1997.
[113] S. Smith, "Flexible filter neighbourhood designation," in Proceedings of 13th International Conference on Pattern Recognition. leee, 1996, pp. 206-212 vol.1. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm? arnumber=546020
[114] Y. Xingfang, H. Yumei, and L. Yan, "An improved SUSAN corner detection algorithm based on adaptive threshold," 2010 2nd International Conference on Signal Processing Systems, pp. V2-613-V2-616, jul 2010. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5555730
[115] S. Stephen, "Determining the position of edges and corners in images." 2014.
[116] N. Dalal and B. Triggs, "Histograms of Oriented Gradients for Human Detection," CVPR '05: Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Volume 1, pp. 886-893, 2005. [Online]. Available: citeulike-article-id:3047126\{\%\}5Cnhttp: //dx.doi.org/10.1109/CVPR.2005.177
[117] J. C. Nunes, S. Guyot, and E. Deléchelle, "Texture analysis based on local analysis of the Bidimensional Empirical Mode Decomposition," Machine Vision and Applications, vol. 16, no. 3, pp. 177-188, may 2005. [Online]. Available: http://link.springer.com/10.1007/s00138-004-0170-5
[118] Z. Liu, H. Wang, and S. Peng, "Texture classification through directional empirical mode decomposition," in Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004. IEEE, 2004, pp. 803-806 Vol.4. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm? arnumber=1333894
[119] M. Unser, "Texture classification and segmentation using wavelet frames." IEEE transactions on image processing : a publication of the IEEE Signal Processing Society, vol. 4, no. 11, pp. 1549-60, jan 1995. [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/18291987
[120] D. R. Martin, C. C. Fowlkes, and J. Malik, "Learning to detect natural image boundaries using local brightness, color, and texture," IEEE transactions on pattern analysis and machine intelligence, vol. 26, no. 5, pp. 530-49, may 2004. [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/15460277
[121] J. Weickert, "Anisotropic diffusion in image processing," Image Rochester NY, vol. 256, no. 3, p. 170, 1998. [Online]. Available: http://citeseerx.ist.psu. edu/viewdoc/download?doi=10.1.1.11.7513\{\&\}rep=rep1\{\&\}type=pdf\{\%\}5Cnhttp: //www.lpi.tel.uva.es/muitic/pim/docus/anisotropic\{_\}diffusion.pdf
[122] G. Vidal-Cassanya, A. Munoz-Barrutia, and M. Unser, "Locally Adaptive Smoothing Method Based on B-splines," fisica.unav.es, no. 3, pp. 1-6, 2006. [Online]. Available: http://fisica.unav.es/Articulos/2006.compIM.pdf
[123] J. Lim, "A Direct Method for Modeling Non-Rigid Motion with Thin Plate Spline," 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), vol. 1, pp. 1196-1202, 2005. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1467402
[124] S. N. Wood, "Thin plate regression splines," Journal of the Royal Statistical

Society: Series B (Statistical Methodology), vol. 65, no. 1, pp. 95-114, feb 2003. [Online]. Available: http://www.blackwell-synergy.com/links/doi/10. 1111\{\%\}2F1467-9868.00374
[125] F. L. Bookstein, "Principal Warps: Thin-Plate Splines and the Decomposition of Deformations," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. II, no. 6, pp. 567-585, 1989.
[126] M. Schulze and J. Pearce, "Linear combinations of morphological operators: the midrange, pseudomedian, and LOCO filters," in IEEE International Conference on Acoustics Speech and Signal Processing, no. 3. IEEE, 1993, pp. 57-60 vol.5. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm? arnumber=319746
[127] W. K. Pratt, T. J. Cooper, and I. Kabir, "Pseudomedian Filter," in Architectures and Algorithms for Digital Image Processing II, F. J. Corbett, Ed., jul 1985, pp. 34-43. [Online]. Available: http://proceedings.spiedigitallibrary.org/proceeding. aspx?articleid=1239275
[128] X. Wang, "On the gradient inverse weighted filter (image processing)," IEEE Transactions on Signal Processing, vol. 40, no. 2, pp. 482-484, 1992. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm? arnumber=124968
[129] M. Zervakis and A. Venetsanopoulos, "Joint optimization in linear and nonlinear image restoration under the presence of different kinds of noise," in Image Processing and its Applications, 1989., Third International Conference on, vol. 1, 1989, pp. 646-650. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs\{_\}all. jsp?arnumber=132208
[130] H. Takeda, S. Farsiu, and P. Milanfar, "Robust kernel regression for restoration and reconstruction of images from sparse noisy data," Proceedings - International Conference on Image Processing, ICIP, pp. 1257-1260, 2006.
[131] R. W. Schafer, "What is a savitzky-golay filter?" IEEE Signal Processing Magazine, vol. 28, no. 4, pp. 111-117, 2011.
[132] ——, "What is a savitzky-golay filter?" IEEE Signal Processing Magazine, vol. 28, no. 4, pp. 111-117, 2011.
[133] T. Xiang and S. Gong, "Spectral clustering with eigenvector selection," Pattern Recognition, vol. 41, no. 3, pp. 1012-1029, 2008.
[134] C. Fowlkes, D. Martin, and J. Malik, "Learning affinity functions for image segmentation: combining patch-based and gradient-based approaches," 2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2003. Proceedings., vol. 2, 2003.
[135] D. R. Martin, C. C. Fowlkes, J. Malik, C. S. Division, and U. C. Berkeley, "Learning to Detect Natural Image Boundaries Using Brightness and Texture," in Advances in Neural Information Processing Systems 2002, S. Becker, S. Thrun, and K. Obermayer, Eds., no. December. MIT Press, 2002, pp. 1279-1286. [Online]. Available: http://papers.nips.cc/paper/ 2217-learning-to-detect-natural-image-boundaries-using-brightness-and-texture. pdf
[136] S. Gupta, P. Arbel, J. Malik, and B. Berkeley, "Perceptual Organization and Recog-
nition of Indoor Scenes from RGB-D Images," The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2013.
[137] C. Fowlkes and J. Malik, "How much does globalization help segmentation?" Berkeley, California, Tech. Rep. July, 2004. [Online]. Available: http://www.eecs. berkeley.edu/Pubs/TechRpts/2004/CSD-04-1340.pdf
[138] E. Boyer, "Edge Detection," pp. 1-18, 2011. [Online]. Available: http: //morpheo.inrialpes.fr/people/Boyer/Teaching/Mosig/
[139] A. Akram and A. Ismail, "Comparison of Edge Detectors," International Journal of Computer Science and Information Technology Research (IJCSITR), vol. 1, no. 1, pp. 16-24, 2013.
[140] S. Xu, L. Han, and L. Zhang, "An Algorithm to Edge Detection Based on SUSAN Filter and Embedded Confidence," Sixth International Conference on Intelligent Systems Design and Applications, vol. 2, pp. 720-723, oct 2006. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm? arnumber=4021752
[141] R. Collins, "CSE486: Harris Corner Detector."
[142] Scikit Learn, "Clustering," 2014.
[143] T. N. Tran, K. Drab, and M. Daszykowski, "Revised DBSCAN algorithm to cluster data with dense adjacent clusters," Chemometrics and Intelligent Laboratory Systems, vol. 120, pp. 92-96, 2013. [Online]. Available: http: //dx.doi.org/10.1016/j.chemolab.2012.11.006
[144] X. Y. Li and L. J. Guo, "Constructing affinity matrix in spectral clustering based on neighbor propagation," Neurocomputing, vol. 97, pp. 125-130, 2012. [Online]. Available: http://dx.doi.org/10.1016/j.neucom.2012.06.023
[145] S. Uw, A. Y. Ng, M. I. Jordan, and Y. Weiss, "On spectral clustering: Analysis and an algorithm," Advances in Neural Information Processing Systems 14, pp. 849-856, 2001.
[146] Saravanan Thirumuruganathan, "Introduction To Mean Shift Algorithm," 2010. [Online]. Available: http://saravananthirumuruganathan.wordpress.com/2010/04/ 01/introduction-to-mean-shift-algorithm/
[147] K. G. Derpanis, "Mean Shift Clustering," 2005.
[148] Y. Li, H. Lu, L. Zhang, S. Yang, and S. Serikawa, "Color Image Segmentation Using Fast Density-Based Clustering Method," in Future Communication, Computing, Control and Management, 2012, pp. 593-598.
[149] R. Manavalan and K. Thangavel, "TRUS image segmentation using morphological operators and DBSCAN clustering," Proceedings of the 2011 World Congress on Information and Communication Technologies, WICT 2011, pp. 898-903, 2011.
[150] Q. Y. Q. Ye, W. G. W. Gao, and W. Z. W. Zeng, "Color image segmentation using density-based clustering," 2003 IEEE International Conference on Acoustics, Speech, and Signal Processing, 2003. Proceedings. (ICASSP '03)., vol. 3, pp. III-345-8 vol.3, 2003. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber= 1199480\{\%\}5Cnpapers2://publication/doi/10.1109/ICASSP.2003.1199480
[151] _-, "Color image segmentation using density-based clustering," 2003 IEEE International Conference on Acoustics, Speech, and Signal Processing, 2003. Pro-
ceedings. (ICASSP '03)., vol. 3, pp. 345-348, 2003.
[152] Scikit Learn, "Gaussian mixture models," 2014. [Online]. Available: http: //scikit-learn.org/stable/modules/mixture.html
[153] S. Borman, "The Expectation Maximization Algorithm A short tutorial," Submitted for publication, vol. 25, no. x, pp. 1-9, 2009. [Online]. Available: http://citeseerx.ist. psu.edu/viewdoc/download?doi=10.1.1.150.8193\{\&\}rep=rep1\{\&\}type=pdf
[154] X. Zhu, "CS838-1 Advanced NLP : The EM Algorithm K-means Clustering," 2007. [Online]. Available: http://pages.cs.wisc.edu/\{~\}jerryzhu/cs838/EM.pdf
[155] T. N. Pappas, "An Adaptive Clustering Algorithm for Image Segmentation," IEEE Transactions on Signal Processing, vol. 40, no. 4, pp. 901-914, 1992.
[156] Z. Lu, Y. Peng, and H. H. Ip, "Combining multiple clusterings using fast simulated annealing," Pattern Recognition Letters, vol. 32, no. 15, pp. 1956-1961, nov 2011. [Online]. Available: http://linkinghub.elsevier.com/retrieve/pii/S0167865511002959
[157] S. Bubeck, M. MeilÄČ, and U. von Luxburg, "How the initialization affects the stability of the Ȯ̇̇-means algorithm," ESAIM: Probability and Statistics, vol. 16, pp. 436-452, sep 2012. [Online]. Available: http://www.esaim-ps.org/10.1051/ps/ 2012013
[158] J. Shi, D. Martin, C. Fowlkes, and E. Sharon, "Graph Based Image Segmentation Tutorial," 2004. [Online]. Available: http://www.cis.upenn.edu/\{~\}jshi/GraphTutorial/
[159] S. Maji, N. K. Vishnoi, and J. Malik, "Biased normalized cuts," Cvpr 2011, pp. 2057-2064, jun 2011. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/ wrapper.htm?arnumber=5995630
[160] H. Zhu, J. Zheng, J. Cai, and N. M. Thalmann, "Object-Level Image Segmentation Using Low Level Cues," IEEE Transactions on Image Processing, vol. 22, no. 10, pp. 4019-4027, 2013.
[161] J. Yuan, E. Bae, X.-C. Tai, and Y. Boykov, "A study on continuous max- flow and min-cut approaches," 2010 IEEE Conference, vol. 7, pp. 2217-2224, 2010.
[162] J. Yuan, E. Bae, X.-c. Tai, and Y. Boykov, "A Continuous Max-Flow Approach to Potts Model," in Computer VisionâĂŞECCV, 2010, pp. 379-392. [Online]. Available: http://link.springer.com/10.1007/978-3-642-15567-3\{_\}28
[163] C. Fowlkes and J. Malik, "Scale-invariant contour completion using conditional random fields," Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1, pp. 1214-1221 Vol. 2, 2005. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1544859
[164] X. Ren and J. Malik, "A Probabilistic Multi-scale Model for Contour Completion Based on Image Statistics," ECCV, pp. 312-327, 2002.
[165] Y. Deng and B. Manjunath, "Unsupervised Segmentation of Color-Texture Regions in Images and Video," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 23, no. 8, pp. 800-810, 2001. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=946985
[166] M. Kass, A. Witkin, and D. Terzopoulos, "Snakes: Active contour models," International Journal of Computer Vision, vol. 1, no. 4, pp. 321-331, 1988.
[167] C. Li, C. Xu, C. Gui, and M. D. Fox, "Level Set Evolution without Re-Initialization: A New Variational Formulation," 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), vol. 1, pp. 430-436,
2005. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm? arnumber=1467299
[168] J. Lie, M. Lysker, and X.-c. Tai, "A VARIANT OF THE LEVEL SET METHOD AND APPLICATIONS TO IMAGE SEGMENTATION," Mathmatics of Computaion, vol. 75, no. 255, pp. 1155-1174, 2006.
[169] K. Museth, D. E. Breen, R. T. Whitaker, S. Mauch, and D. Johnson, "Algorithms for Interactive Editing of Level Set Models," Computer Graphics Forum, vol. 24, no. 4, pp. 821-841, dec 2005. [Online]. Available: http: //doi.wiley.com/10.1111/j.1467-8659.2005.00904.x
[170] N. Paragiosl and R. Deriche, "Coupled Geodesic Active Regions for Image Segmentation : A Level Set Approach," in Computer Vision, D. Vernon, Ed. Springer Berlin / Heidelberg, 2000, pp. 224-240.
[171] A. Belaid, D. Boukerroui, Y. Maingourd, and J.-F. Lerallut, "Phase based level set segmentation of ultrasound images," Information Technology and Applications in Biomedicine, Larnaca : Cyprus, jan 2009. [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/21216695
[172] S. Osher and N. Paragios, Geometric Level Set Methods in Imaging, Vision, and Graphics, S. Osher and N. Paragios, Eds. Springer, 2003. [Online]. Available: http://mywebpages.comcast.net/aparagio/book.htmhttp://books. google.com/books?id=IHmVozwr9cQC\{\&\}|pg=PR15\{\&\}ots=XEhaVEJisy\{\&\}dq= levelsetimaging\{\&\}|r\{\&\}pg=PP1\{\#\}v=onepage\{\&\}q=levelset\{\&\}f=false
[173] N. Paragios, "A level set approach for shape-driven segmentation and tracking of the left ventricle." IEEE transactions on medical imaging, vol. 22, no. 6, pp. 773-6, jun 2003. [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/12872953
[174] J. Lie, M. Lysaker, and X.-C. Tai, "A binary level set model and some applications to Mumford-Shah image segmentation." IEEE transactions on image processing : a publication of the IEEE Signal Processing Society, vol. 15, no. 5, pp. 1171-81, may 2006. [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/16671298
[175] L. Bertelli, B. Sumengen, B. S. Manjunath, and F. Gibou, "A variational framework for multiregion pairwise-similarity-based image segmentation," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 30, no. 8, pp. 1400-1414, 2008.
[176] N. Salman, "Image Segmentation Based on Watershed and Edge Detection Techniques," The International Arab Journal of Information Technology, vol. 3, no. 2, pp. 104-110, 2006.
[177] S. Olsen and B. Gooch, "Image simplification and vectorization," Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Non-Photorealistic Animation and Rendering - NPAR '11, p. 65, 2011. [Online]. Available: http://dl.acm.org/ citation.cfm?doid=2024676.2024687
[178] P. Lindstrom, "Image-Driven Simplification," ACMTransactions on Graphics, vol. 19, no. 3, 2000.
[179] P. C. Symposium, "IMAGE SIMPLIFICATION BY FREQUENCY-SELECTIVE MEANS FILTERING Institut f Âl ur Nachrichtentechnik RWTH Aachen University Aachen , Germany," Work, pp. 126-129, 2010.
[180] F. Meyer, "Levelings, Image Simplification Filters for Segmentation," Journal of Mathematical Imaging and Vision, vol. 20, no. 1/2, pp. 59-72, 2004. [Online].

Available: http://link.springer.com/10.1023/B:JMIV.0000011319.21884.39
[181] H. Gao, W.-c. Siu, and C.-h. Hou, "Improved techniques for automatic image segmentation," IEEE Transactions on Circuits and Systems for Video Technology, vol. 11, no. 12, pp. 1273-1280, 2001. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=974681
[182] W. T. Freeman, T. R. Jones, and E. C. Pasztor, "Example-Based Super-Resolution," IEEE, no. April, pp. 56-65, 2002.
[183] A. Dobson, "Introduction to generalized linear models," 2013. [Online]. Available: https://onlinecourses.science.psu.edu/stat504/node/216
[184] R. Mora, "Two stage least squares regression (2SLS)," 2013. [Online]. Available: http://www.eco.uc3m.es/\{~\}ricmora/MEI/materials/Session\{_\}09\{_\}2SLS.pdf
[185] G. Rodriguez, "Logit Models for Binary Data," in Unknown, 2007.
[186] D. Cook, P. Dixon, W. M. Duckworth, M. S. Kaiser, K. Koehler, W. Q. Meeker, and W. R. Stephenson, "Binary Response and Logistic Regression Analysis," in Unknown, 2000. [Online]. Available: http://www.public.iastate.edu/\{~\}stat415/ \{\%\}5Cnpapers2://publication/uuid/81EA8EA6-4236-470B-AA46-8EC3953EAA32
[187] R. Tibshirani, "Regularization: Ridge Regression and the LASSO," 2007. [Online]. Available: http://statweb.stanford.edu/\{~\}tibs/sta305files/Rudyregularization.pdf
[188] Y. Sun, "Ridge Regression , LASSO and Elastic Net," 2014. [Online]. Available: https://www.slideshare.net/ShangxuanZhang/ ridge-regression-lasso-and-elastic-net
[189] T. Hastie, R. Tibshirani, and M. Wainwright, Statistical Learning with Sparsity: The Lasso and Generalizations. Boca Raton, FL: CRC Press, 2015. [Online]. Available: https://web.stanford.edu/\{~\}hastie/StatLearnSparsity\{_\}files/ SLS\{_\}corrected\{_\}1.4.16.pdf
[190] Nicoguaro, "L1L2," 2017. [Online]. Available: https://commons.wikimedia.org/wiki/ File:L1\{_\}and\{_\}L2\{_\}balls.svg
[191] H. Zou and T. Hastie, "Regularization and Variable Selection via the Elastic Net," 2005. [Online]. Available: https://web.stanford.edu/\{~\}hastie/TALKS/enet\{_\}talk.pdf
[192] R. Goebel, Machine Learning and Data Mining in Pattern Recognition, ser. Lecture Notes in Computer Science, P. Perner, Ed. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, vol. 6871. [Online]. Available: http: //www.springerlink.com/index/10.1007/978-3-642-23199-5
[193] D. Liu, Y. Xiong, K. Pulli, and L. Shapiro, "Estimating Image Segmentation Difficulty Dingding," in Machine Learning and Data Mining in Pattern Recognition, ser. Lecture Notes in Computer Science, P. Perner, Ed. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, vol. 6871, pp. 484-495. [Online]. Available: http://link.springer.com/chapter/10.1007/978-3-642-23199-5\{_\}36
[194] J. Siekmann, Learning Theory and Kernel Machines, ser. Lecture Notes in Computer Science, B. Schölkopf and M. K. Warmuth, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2003, vol. 2777. [Online]. Available: http://link.springer.com/10.1007/b12006
[195] J. Feng and N. Simon, "Sparse-Input Neural Networks for High-dimensional Nonparametric Regression and Classification," University ofWashington, nov 2017. [Online]. Available: http://arxiv.org/abs/1711.07592
[196] L. Yu and H. Liu, "Feature Selection for High-Dimensional Data: A Fast Correlation-Based Filter Solution," International Conference on Machine Learning (ICML), pp. 1-8, 2003. [Online]. Available: http://www.aaai.org/Papers/ICML/2003/ ICML03-111.pdf
[197] S. Saitta, "Standardization vs. normalization," 2007. [Online]. Available: http://www.dataminingblog.com/standardization-vs-normalization/
[198] E. Walker and A. S. Nowacki, "Understanding equivalence and noninferiority testing," Journal of General Internal Medicine, vol. 26, no. 2, pp. 192-196, 2011.
[199] A. Schnell, "The Difference Between Relative Risk and Odds Ratios," pp. 1-5, 2017. [Online]. Available: http://www.theanalysisfactor.com/ the-difference-between-relative-risk-and-odds-ratios/
[200] S. Kanchanaraksa, "Estimating Risk," 2008.
[201] M. Wang, "Generalized Estimating Equations in Longitudinal Data Analysis: A Review and Recent Developments," Advances in Statistics, vol. 2014, pp. 1-11, 2014. [Online]. Available: http://www.hindawi.com/journals/as/2014/303728/
[202] C. Zaiontz, "Intraclass Correlation," 2013. [Online]. Available: http://www. real-statistics.com/reliability/intraclass-correlation/
[203] S. Rubio, E. Díaz, J. Martín, and J. M. Puente, "Evaluation of Subjective Mental Workload: A Comparison of SWAT, NASA-TLX, and Workload Profile Methods," Applied Psychology, vol. 53, no. 1, pp. 61-86, 2004.
[204] A. Bangor, P. T. Kortum, and J. T. Miller, "An Empirical Evaluation of the System Usability Scale," International Journal of Human-Computer Interaction, vol. 24, no. 6, pp. 574-594, 2008. [Online]. Available: http: //www.tandfonline.com/doi/abs/10.1080/10447310802205776
[205] H. T. Lawless and H. Heymann, "Preference Testing," in Sensory Evaluation of Food, 2010, pp. 303-324. [Online]. Available: http://link.springer.com/10.1007/ 978-1-4419-6488-5http://link.springer.com/10.1007/978-1-4419-6488-5\{_\}13
[206] D. Huttenlocher, G. Klanderman, and W. Rucklidge, "Comparing images using the Hausdorff distance," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 15, no. 9, pp. 850-863, 1993. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=232073
[207] A. Y. Yang, J. Wright, Y. Ma, and S. S. Sastry, "Unsupervised segmentation of natural images via lossy data compression," Computer Vision and Image Understanding, vol. 110, no. 2, pp. 212-225, may 2008. [Online]. Available: http://linkinghub.elsevier.com/retrieve/pii/S1077314207001300
[208] J. Freixenet, X. Mu, D. Raba, J. Marti, and X. Cuf, "Yet Another Survey on Image Segmentation: Region and Boundary Information Integration," in ECCV, ser. Lecture Notes in Computer Science, A. Heyden, G. Sparr, M. Nielsen, and P. Johansen, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, apr 2002, vol. 2352, pp. 408-422. [Online]. Available: http: //link.springer.com/10.1007/3-540-47977-5
[209] T. Malisiewicz and A. A. Efros, "Improving Spatial Support for Objects via Multiple Segmentations," in Robotics Commons, 2007.
[210] R. Unnikrishnan, C. Pantofaru, and M. Hebert, "Toward objective evaluation of image segmentation algorithms." IEEE transactions on pattern analysis and
machine intelligence, vol. 29, no. 6, pp. 929-44, jun 2007. [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/17431294
[211] R. Unnikrishnan and M. Hebert, "Measures of Similarity," 2005 Seventh IEEE Workshops on Applications of Computer Vision (WACV/MOTION'05) - Volume 1, pp. 394-394, jan 2005. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/ epic03/wrapper.htm?arnumber=4129508
[212] N. X. Vinh, J. Epps, J. Bailey, and M. Meila, "Information Theoretic Measures for Clusterings Comparison: Variants, Properties, Normalization and Correction for Chance," Journal of Machine Learning Research, vol. 11, pp. 2837-2854, 2010.
[213] Y. Y. Yao, "Entropy Measures, Maximum Entropy Principle and Emerging Applications," in Entropy Measures, Maximum Entropy Principle and Emerging Applications Studies in Fuzziness and Soft Computing, ser. Studies in Fuzziness and Soft Computing. Berlin, Heidelberg: Springer Berlin Heidelberg, 2003, vol. 119, pp. 115-136. [Online]. Available: http://www.springerlink.com/index/10.1007/ 978-3-540-36212-8
[214] A. Rosenberg and J. Hirschberg, "V-measure: A conditional entropy-based external cluster evaluation measure," Computational Linguistics, vol. 1, no. June, pp. 410-420, 2007. [Online]. Available: http://acl.ldc.upenn.edu/D/D07/D07-1043.pdf
[215] M. MeilÇŐ, "Comparing clusterings," in Proceedings of the 22nd international conference on Machine learning - ICML '05. New York, New York, USA: ACM Press, 2005, pp. 577-584. [Online]. Available: http://portal.acm.org/citation.cfm? doid=1102351.1102424
[216] M. MeilÄČ, "Comparing clusteringsâĂŤTan information based distance," Journal of Multivariate Analysis, vol. 98, no. 5, pp. 873-895, may 2007. [Online]. Available: http://linkinghub.elsevier.com/retrieve/pii/S0047259X06002016
[217] -, "Comparing Clusterings by the Variation of Information," in Learning Theory and Kernel Machines, ser. Lecture Notes in Computer Science, B. Schölkopf and M. K. Warmuth, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2003, vol. 2777, pp. 173-187.
[218] B. Peng and L. Zhang, "Evaluation of Image Segmentation Quality by Adaptive Ground Truth Composition," in Computer Vision âĂŞ ECCV 2012, ser. Lecture Notes in Computer Science, A. Fitzgibbon, S. Lazebnik, P. Perona, Y. Sato, and C. Schmid, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, vol. 7574, pp. 287-300.
[219] R. Unnikrishnan, C. Pantofaru, and M. Hebert, "A Measure for Objective Evaluation of Image Segmentation Algorithms," in 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Workshops, vol. 3, no. c. IEEE, 2005, pp. 34-34. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1565332
[220] L. Hubert and P. Arabie, "Comparing partitions," Journal of Classification, vol. 2, no. 1, pp. 193-218, dec 1985. [Online]. Available: http://link.springer.com/10.1007/ BF01908075
[221] C. Carpineto and G. Romano, "Consensus clustering based on a new Probabilistic Rand Index with application to subtopic retrieval." IEEE transactions on pattern analysis and machine intelligence, vol. 34, no. 12, pp. 2315-26, dec 2012.
[Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/22450816
[222] D. M. W. Powers, "Evaluation: From Precision, Recall and F-Factor to ROC, Informedness, Markedness \& Correlation," Flinders University Academic Commons, no. December, 2007. [Online]. Available: http://hdl.handle.net/2328/ 27165
[223] I. Abdou and W. Pratt, "Quantitative design and evaluation of enhancement/thresholding edge detectors," Proceedings of the IEEE, vol. 67, no. 5, pp. 753-763, 1979. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm? arnumber=1455594
[224] K. Bowyer, C. Kranenburg, and S. Dougherty, "Edge detector evaluation using empirical ROC curves," Proceedings. 1999 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (Cat. No PR00149), pp. 354-359, 1999. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm? arnumber=786963
[225] S. A. Macskassy and F. Provost, "Confidence Bands for ROC Curves : Methods and an Empirical Study," in Proceedings of the First Workshop on ROC Analysis in Al. August 2004, no. Aug, 2004. [Online]. Available: http://hdl.handle.net/2451/27802
[226] P. A. Flach, "The many faces of ROC analysis in machine learning," in The TwentyFirst International Conference on Machine Learning, 2004. [Online]. Available: http://www.rduin.nl/presentations/ROCTutorialPeterFlach/ROCtutorialPartl.pdf
[227] Kakau, "Receiver Operating Characteristic," 2003. [Online]. Available: http: //commons.wikimedia.org/wiki/File:Receiver\{_\}Operating\{_\}Characteristic.png
[228] B. W. Matthews, "Comparison of the predicted and observed secondary structure of T4 phage lysozyme," BBA - Protein Structure, vol. 405, no. 2, pp. 442-451, 1975.
[229] J. Pont-Tuset and F. Marques, "Measures and Meta-Measures for the Supervised Evaluation of Image Segmentation," in 2013 IEEE Conference on Computer Vision and Pattern Recognition. IEEE, jun 2013, pp. 2131-2138. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6619121
[230] C. J. V. Rijsbergen, Information Retrieval, 2nd ed. Information Retrieval Group, University of Glasgow, 1979. [Online]. Available: http://www.dcs.gla.ac.uk/Keith/ Preface.html
[231] A. Jepson, "Introduction to Image Segmentation." [Online]. Available: http: //www.cs.toronto.edu/\{~\}jepson/researchlSeg.html
[232] Walber, "Precision Recall," 2014. [Online]. Available: https://en.wikipedia.org/wiki/ Precision\{_\}and\{_\}recall\{\#\}/media/File:Precisionrecall.svg
[233] S. I. Inc., "SAS Plot Statement," 1999. [Online]. Available: http://support.sas.com/documentation/cdl/en/statug/63033/HTML/default/ viewer.htm\{\#\}statug\{_\}boxplot\{_\}sect012.htm
[234] T. Ferro and D. Pawluk, "Providing Dynamic Access to Electronic Tactile Diagrams," in Universal Access in HumanâĂŞComputer Interaction. Designing Novel Interactions. UAHCI 2017. Lecture Notes in Computer Science, ser. Lecture Notes in Computer Science, M. Antona and C. Stephanidis, Eds., vol. 10278. Cham: Springer International Publishing, 2017, pp. 269282. [Online]. Available: http://link.springer.com/10.1007/978-3-319-58703-5http:
//link.springer.com/10.1007/978-3-319-58703-5\{_\}20
[235] T. J. Ferro and D. T. V. Pawluk, "Automatic image conversion to tactile graphic," in Proceedings of the 15th International ACM SIGACCESS Conference on Computers and Accessibility - ASSETS '13. New York, New York, USA: ACM Press, 2013. [Online]. Available: http://dl.acm.org/citation.cfm?doid=2513383. 2513406
[236] A. M. Nguyen, T. J. Ferro, and D. T. Pawluk, "Effectiveness of Using Local Cues to Indicate Perspective in Tactile Diagrams," JVIB, vol. In Review, 2018.
[237] I. Linville, T. Ferro, and D. Pawluk, "The effectiveness of adding local cues to indicate perspective in tactile diagrams and software support for generating these diagrams." in 2018 Virginia AER Conference - Achieving Independence in All Walks of Life. In Review, 2018.
[238] S. Kandalam, T. Ferro, and D. Pawluk, "The Development and Assessment of Crayons Producing Textured Lines," JVIB, vol. In Review, 2018.
[239] R. Rastogi, "A study towards development of an automated haptic user interface for individuals who are blind or visually impaired," Ph.D. dissertation, Virginia Commonwealth University, 2012.
[240] D. Abinum, "Students who are blind face greater social challenges on campus," sep 2015. [Online]. Available: http://college.usatoday.com/2015/09/21/ college-students-blind-challenges/
[241] Adobe, "Photoshop." [Online]. Available: http://www.photoshop.com/
[242] A. N. Albatineh, M. Niewiadomska-Bugaj, and D. Mihalko, "On Similarity Indices and Correction for Chance Agreement," Journal of Classification, vol. 23, no. 2, pp. 301-313, sep 2006. [Online]. Available: http://link.springer.com/10.1007/ s00357-006-0017-z
[243] a. Almansa, a. Desolneux, and S. Vamech, "Vanishing point detection without any a priori information," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 25, no. 4, pp. 502-507, 2003.
[244] C. J. Alpert and S.-z. Yao, "Spectral Partitioning: The More Eigenvectors, The Better," in 32nd Design Automation Conference. New York, New York, USA: ACM, dec 1995, pp. 195-200. [Online]. Available: http://portal.acm.org/citation.cfm?doid=217474.217529http://ieeexplore. ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1586701
[245] F. a. Andal, G. Taubin, and S. Goldenstein, "Vanishing Point Detection by Segment Clustering on the Projective Space," in ECCV 2010: Trends and Topics in Computer Vision, 2010, pp. 324-337.
[246] M. Antona and C. Stephanidis, Universal Access in HumanâĂŞComputer Interaction. Designing Novel Interactions, ser. Lecture Notes in Computer Science, M. Antona and C. Stephanidis, Eds. Cham: Springer International Publishing, 2017, vol. 10278. [Online]. Available: http://link.springer.com/10.1007/ 978-3-319-58703-5http://link.springer.com/10.1007/978-3-319-58703-5\{_\}20
[247] APH, "Teacher Training." [Online]. Available: http://www.aph.org/training/
[248] P. Arbelaez, B. Hariharan, S. Gupta, L. Bourdev, and J. Malik, "Semantic segmentation using regions and parts," in 2012 IEEE Conference on Computer Vision and Pattern Recognition. IEEE, jun 2012, pp. 3378-3385. [Online].

Available: http://ieeexplore.ieee.org/xpls/abs\{_\}all.jsp?arnumber=6248077http:// ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6248077
[249] M. W. Asghar, K. E. Barner, and S. Member, "Nonlinear Multiresolution Techniques with Applications to Scientific Visualization in a Haptic Environment," IEEE Transactions on Visualization and Computer Graphics, vol. 7, no. 1, pp. 76-93, 2001.
[250] J. Barron and J. Malik, "Color Constancy, Intrinsic Images, and Shape Estimation," ECCV, 2012. [Online]. Available: http://www.cs.berkeley.edu/ \{~\}barron/BarronMalikECCV2012.pdf
[251] N. Bi, Q. Sun, D. Huang, Z. Yang, and J. Huang, "Robust image watermarking based on multiband wavelets and empirical mode decomposition." IEEE transactions on image processing : a publication of the IEEE Signal Processing Society, vol. 16, no. 8, pp. 1956-66, aug 2007. [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/17688201
[252] T. Brox, L. Bourdev, S. Maji, and J. Malik, "Object segmentation by alignment of poselet activations to image contours," Cvpr 2011, pp. 2225-2232, jun 2011. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm? arnumber=5995659
[253] T. Carron and P. Lambert, "Color edge detector using jointly hue, saturation and intensity," in Proceedings of 1st International Conference on Image Processing, vol. 3. IEEE Comput. Soc. Press, 1994, pp. 977-981. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=413699
[254] J. Chen and N. Takagi, "A pattern recognition method for automating tactile graphics translation from hand-drawn maps," Proceedings - 2013 IEEE International Conference on Systems, Man, and Cybernetics, SMC 2013, vol. 4173, no. c, pp. 4173-4178, 2013.
[255] C. M. Christoudias, B. Georgescu, and P. Meer, "Synergism in low level vision," in Pattern Recognition, 2002. Proceedings. 16th International Conference on, vol. 4, no. 1. IEEE Comput. Soc, 2002, pp. 150-155. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1047421
[256] R. Cipolla, T. Drummond, and D. Robertson, "Camera calibration from vanishing points in images of architectural scenes," Review Literature And Arts Of The Americas, vol. 2, pp. 382-391, 1999. [Online]. Available: http://citeseerx.ist.psu. edu/viewdoc/download?doi=10.1.1.16.946\{\&\}rep=rep1\{\&\}type=pdf
[257] T. Cour and F. Benezit, "Spectral Segmentation with Multiscale Graph Decomposition," in 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), vol. 2. IEEE, 2005, pp. 11241131. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm? arnumber=1467569
[258] E. Davis, Machine Vision, 3rd ed. San Francisco: Morgan Kaufmann, 2005.
[259] A. Desolneux, L. Moisan, and J.-M. Morel, From Gestalt Theory to Image Analysis, ser. Interdisciplinary Applied Mathematics. New York, NY: Springer New York, 2008, vol. 34, no. July. [Online]. Available: http://scholar.google.com/scholar?hl=en\{\&\}btnG=Search\{\&\}q=intitle: From+Gestalt+Theory+to+Image+Analysis\{\#\}5http://w3.mi.parisdescartes.fr/ map5/Publis/2006-09.pdfhttp://link.springer.com/10.1007/978-0-387-74378-3
[260] P. Dollar and S. Belongie, "Supervised Learning of Edges and Object Boundaries," in 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Volume 2 (CVPR'06), vol. 2. IEEE, 2006, pp. 1964-1971. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper. htm?arnumber=1640993
[261] A. Ecker, A. D. Jepson, and K. N. Kutulakos, "Semidefinite Programming Heuristics for Surface Reconstruction Ambiguities," ECCV 2008, pp. 127-140, 2008.
[262] F. J. Estrada, "ADVANCES IN COMPUTATIONAL IMAGE SEGMENTATION AND PERCEPTUAL GROUPING," Ph.D. dissertation, University of Toronto, 2005.
[263] M. Felsberg, "Optical flow estimation from monogenic phase," Complex Motion, pp. 1-13, 2007. [Online]. Available: http://link.springer.com/chapter/10.1007/ 978-3-540-69866-1\{_1
[264] M. Felsberg and G. Sommer, "The monogenic signal The Monogenic Signal," IEEE Transactions on Signal Processing, vol. 49, no. 12, pp. 3136-3144, 2001.
[265] P. Felzenszwalb and D. Huttenlocher, "Image segmentation using local variation," in Proceedings. 1998 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (Cat. No.98CB36231). IEEE Comput. Soc, 1998, pp. 98-104. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm? arnumber=698594
[266] P. Felzenszwalb and D. McAllester, "A Min-Cover Approach for Finding Salient Curves," in 2006 Conference on Computer Vision and Pattern Recognition Workshop (CVPRW'06). IEEE, 2006, pp. 185-185. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1640633
[267] C. Feng, F. Deng, and V. R. Kamat, "Semi-Automatic 3D Reconstruction of Piecewise Planar," 10th International Conference on Construction Applications of Virtual Reality, pp. 1-9, 2010.
[268] F. Gallegos-Funes, V. Ponomaryov, S. Sadovnychiy, and L. Nino-de Rivera, "Median M-type K-nearest neighbour (MM-KNN) filter to remove impulse noise from corrupted images," Electronics Letters, vol. 38, no. 15, p. 786, 2002. [Online]. Available: http://digital-library.theiet.org/content/journals/10.1049/el\{_\}20020567
[269] D. Gevorkian, K. Egiazarian, and J. Astola, "Modified K-nearest neighbour filters for simple implementation," in 2000 IEEE International Symposium on Circuits and Systems. Emerging Technologies for the 21st Century. Proceedings (IEEE Cat No.00CH36353), vol. 4. Presses Polytech. Univ. Romandes, 2000, pp. 565-568. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper. htm?arnumber=858814
[270] GIMP, "GIMP." [Online]. Available: http://www.gimp.org/
[271] L. Grady, "Random walks for image segmentation." IEEE transactions on pattern analysis and machine intelligence, vol. 28, no. 11, pp. 1768-83, nov 2006. [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/17063682
[272] R. Grompone Von Gioi, J. Jakubowicz, J. M. Morel, and G. Randall, "LSD: A fast line segment detector with a false detection control," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 32, no. 4, pp. 722-732, 2010.
[273] D. S. Guru, B. H. Shekar, and P. Nagabhushan, "A simple and robust line detection algorithm based on small eigenvalue analysis," Pattern Recognition Letters, vol. 25,
no. 1, pp. 1-13, 2004.
[274] R. M. Haralick, "Zero crossing of second directional derivative edge operator," in 1982 Technical Symposium East, A. Rosenfeld, Ed., nov 1982, pp. 91101. [Online]. Available: http://proceedings.spiedigitallibrary.org/proceeding.aspx? articleid=1233359
[275] R. Hartley and A. Zisserman, Multiple View Geometry in Computer Vision, 2nd ed. Cambridge: Cambridge University Press, 2000.
[276] P. C. Headley and D. T. V. Pawluk, "Roughness perception of textures on a haptic matrix display," 2011 IEEE World Haptics Conference, WHC 2011, pp. 221-226, 2011.
[277] M. a. Heller and J. M. Kennedy, "Perspective taking, pictures, and the blind." Perception \& psychophysics, vol. 48, no. 5, pp. 459-466, 1990.
[278] M. a. Heller, D. D. Brackett, E. Scroggs, H. Steffen, K. Heatherly, and S. Salik, "Tangible pictures: Viewpoint effects and linear perspective in visually impaired people," Perception, vol. 31, no. 6, pp. 747-769, 2002.
[279] M. a. Heller, M. McCarthy, and A. Clark, "Pattern perception and pictures for the blind," Psicologica, vol. 26, no. 1, pp. 161-171, 2005.
[280] P. R. Hill, C. Nishan Canagarajah, and D. R. Bull, "Image segmentation using a texture gradient based watershed transform," IEEE Transactions on Image Processing, vol. 12, no. 12, pp. 1618-1633, 2003.
[281] V. Hlaváč, "Classifier performance evaluation," Ppt, 2012. [Online]. Available: http://cmp.felk.cvut.cz/Ëœhlavac
[282] D. Hutchison and J. C. Mitchell, Computer Vision âĂŞ ECCV 2012, ser. Lecture Notes in Computer Science, A. Fitzgibbon, S. Lazebnik, P. Perona, Y. Sato, and C. Schmid, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, vol. 7574. [Online]. Available: http://www.springerlink.com/index/10.1007/ 978-3-642-33712-3
[283] S. Inge, "Using SVD for some fitting problems," 2009.
[284] Z. Jin-yu, C. Yan, and H. Xian-xiang, "Edge detection of images based on improved Sobel operator and genetic algorithms," 2009 International Conference on Image Analysis and Signal Processing, no. 3, pp. 31-35, 2009. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5054605
[285] N. Kanopoulos, N. Vasanthavada, and R. Baker, "Design of an image edge detection filter using the Sobel operator," IEEE Journal of SolidState Circuits, vol. 23, no. 2, pp. 358-367, apr 1988. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=996
[286] Y. Keselman and E. Micheli-Tzanakou, "Extraction and characterization of regions of interest in biomedical images," Proceedings. 1998 IEEE International Conference on Information Technology Applications in Biomedicine, ITAB '98 (Cat. No.98EX188), pp. 87-90, 1998. [Online]. Available: http: //ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=674683
[287] H. Kong, J.-Y. Audibert, and J. Ponce, "Vanishing point detection for road detection," IEEE Conference on Computer Vision and Pattern Recognition (2008), no. 3, pp. 96-103, 2009. [Online]. Available: http://eprints.pascal-network.org/ archive/00006913/
[288] S. Konishi, A. Yuille, and J. Coughlan, "Fundamental bounds on edge detection: an information theoretic evaluation of different edge cues," in Proceedings. 1999 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (Cat. No PR00149). IEEE Comput. Soc, 1999, pp. 573-579. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=786996
[289] M. Köppen, K. Yoshida, and P. Valle, "Gestalt theory in image processing: A discussion paper," Three-Rivers Workshop on Soft Computing in Industrial Applications, 2007. [Online]. Available: http://science.mkoeppen.com/science/ uploads/Publications/smcia07.pdf
[290] a. Kraskov, H. Stögbauer, R. G. Andrzejak, and P. Grassberger, "Hierarchical clustering using mutual information," Europhysics Letters (EPL), vol. 70, no. 2, pp. 278-284, apr 2005. [Online]. Available: http://stacks.iop.org/0295-5075/70/i=2/a= 278?key=crossref.4b80db2e1ce6b59ac95a3a163d16c2ed
[291] S. J. Lederman and R. L. Klatzky, "Relative availability of surface and object properties during early haptic processing." Journal of experimental psychology. Human perception and performance, vol. 23, no. 6, pp. 1680-1707, 1997.
[292] J. Lee, R. Haralick, and L. Shapiro, "Morphologic edge detection," IEEE Journal on Robotics and Automation, vol. 3, no. 2, pp. 142-156, apr 1987. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1087088
[293] J. M. Loomis, "A model of character recognition and legibility." Journal of experimental psychology. Human perception and performance, vol. 16, no. 1, pp. 106120, 1990.
[294] -_, "Analysis of tactile and visual confusion matrices." Perception \& psychophysics, vol. 31, no. 1, pp. 41-52, 1982.
[295] ——, "On the tangibility of letters and braille." Perception \& psychophysics, vol. 29, no. 1, pp. 37-46, 1981.
[296] J. M. Loomis, R. L. Klatzky, and S. J. Lederman, "Similarity of tactual and visual picture recognition with limited field of view." Perception, vol. 20, no. 2, pp. 167177, 1991.
[297] J. M. Loomis, R. L. Klatzky, and N. A. Giudice, "Sensory substitution of vision: Importance of perceptual and cognitive processing," in Assistive Technology for Blindness and Low Vision, 2012, pp. 162-191.
[298] Z. Luo, "Clustering under spatial contiguity constraints: a penalized K-means method," IT Workship on Setection, Estimation, Classification and Imaging, 1999. [Online]. Available: http://www.ifp.illinois.edu/itw-deci/psfiles/luo.ps
[299] J. Mairal, M. Leordeanu, F. Bach, M. Hebert, and J. Ponce, "Discriminative Sparse Image Models for Class-Specific Edge Detection and Image Interpretation," ECCV 2008, pp. 43-56, 2008.
[300] M. Maire, P. Arbelaez, C. Fowlkes, and J. Malik, "Using contours to detect and localize junctions in natural images," in 2008 IEEE Conference on Computer Vision and Pattern Recognition. IEEE, jun 2008, pp. 1-8. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=4587420
[301] J. Malik and R. Rosenholtz, "Computing local surface orientation and shape from texture for curved surfaces," International Journal of Computer Vision, vol. 23, no. 2, pp. 149-168, 1997. [On-
line]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber= 937659http://www.springerlink.com/index/u831115t42065231.pdf
[302] D. Martin, "Evaluating Segmentation," 2004.
[303] P. B. L. Meijer, "An experimental system for auditory image representations," IEEE Transactions on Biomedical Engineering, vol. 39, no. 2, pp. 112-121, 1992.
[304] L. Miao and S. Peng, "Perspective Rectification of Document Images Based on Morphology," 2006 International Conference on Computational Intelligence and Security, pp. 1805-1808, 2006. [Online]. Available: http: //ieeexplore.ieee.org/pdocs/epic03/wrapper.htm?arnumber=4076280
[305] M. C. Morrone and D. C. Burr, "Feature detection in human vision: a phase-dependent energy model." Proceedings of the Royal Society of London. Series B, Containing papers of a Biological character. Royal Society (Great Britain), vol. 235, no. 1280, pp. 221-45, dec 1988. [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/2907382
[306] D. Mumford and J. Shah, "Optimal approximations by piecewise smooth functions and associated variational problems," Communications on pure and applied ..., vol. XLII, pp. 577-685, 2006. [Online]. Available: http: //onlinelibrary.wiley.com/doi/10.1002/cpa.3160420503/abstract
[307] S. C. W. Ong and S. Ranganath, "Automatic sign language analysis: A survey and the future beyond lexical meaning," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 27, no. 6, pp. 873-891, 2005.
[308] OPI Online Course, "Systems CIELAB and CIELUV," 2018. [Online]. Available: http://www.optique-ingenieur.org/en/courses/OPI\{_\}ang\{\}M07\{_\}C02/ co/Contenu\{_308.html
[309] A. Paiement, M. Mirmehdi, X. Xie, and M. Hamilton, "Object Modelling from Sparse and Misaligned 3D and 4D data," 2014. [Online]. Available: https://vilab.blogs.ilrt.org/?p=588
[310] T. N. Pappas, V. C. Tarter, A. G. Seward, B. Genzer, K. Gourgey, and I. Kretzschmar, "Perceptual Dimensions for a Dynamic Tactile Display," in Human Vision and Electronic Imaging XIV, A. J. Woods, I. E. McDowall, J. A. Beraldin, B. E. Rogowitz, K. Börner, N. Kehtarnavaz, C. A. Bouman, R. Eschbach, S. P. Farnand, J. T. Astola, K. Berkner, B. G. Rodricks, E. J. Delp III, R. Schettini, F. Truchetet, E. Bodegom, K. S. Niel, D. P. Casasent, R. Rejaie, R. Creutzburg, M. Rabbani, D. Akopian, K. D. Mayer-Patel, E. L. Hall, D. Fofi, V. Nguyen, O. Laligant, R. C. Jain, M. Dolinsky, R. L. Stevenson, J. Dittmann, S. E. Süsstrunk, L. Likforman-Sulem, K. O. Egiazarian, F. Gaykema, G. G. Marcu, E. L. Miller, M. F. Carlsohn, J. Park, T. N. Pappas, G. S. Cheok, N. S. Holliman, J. O. Merritt, M. McCarthy, I. Pollak, S. Tominaga, N. M. Nasrabadi, N. D. Memon, S. Santini, J. Röning, P. W. Wong, S. A. Rizvi, U. Neuschaefer-Rube, and A. Rizzi, Eds., vol. 7240, feb 2009, pp. 72 400K-72 400K-12. [Online]. Available: http://spiedigitallibrary.org/proceeding.aspx?doi=10.1117/12.817182
[311] D. Pawluk, "Grant Proposal," pp. 1-15, 2012.
[312] P. Perona and J. Malik, "Scale-space and edge detection using anisotropic diffusion," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 12, no. 7, pp. 629-639, jul 1990. [Online]. Available: http://ieeexplore.ieee.org/

Ipdocs/epic03/wrapper.htm?arnumber=56205
[313] ——, "Detecting and localizing edges composed of steps, peaks and roofs," in [1990] Proceedings Third International Conference on Computer Vision. IEEE Comput. Soc. Press, 1990, pp. 52-57. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=139492
[314] C. Perra, F. Massidda, and D. Giusto, "Image blockiness evaluation based on Sobel operator," in IEEE International Conference on Image Processing 2005. IEEE, 2005, pp. I-389. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/ epic03/wrapper.htm?arnumber=1529769
[315] R. Reichart and A. Rappoport, "The NVI Clustering Evaluation Measure," in CoNLL '09 Proceedings of the Thirteenth Conference on Computational Natural Language Learning, no. June, 2009, pp. 165-173.
[316] X. Ren, "Multi-scale Improves Boundary Detection," ECCV 2008, pp. 533-545, 2008.
[317] J. Robinson, "Optimal Detection of Blurred Edges," Object recognition supported by user interaction for service robots, vol. 3, no. x, pp. 831-834, 2002. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm? arnumber=1048152
[318] C. Simon, S. Cho, and I. K. Park, "Fast and Robust Perspective Rectification of Document Images on a Smartphone," IEEE Conference on Computer Vision and Pattern Recognition Workshops, pp. 2-3, 2014.
[319] Springer-Verlag, Future Communication, Computing, Control and Management, Y. Zhang, Ed. Heidelberg: Springer-Verlag, 2012, vol. 141. [Online]. Available: http://link.springer.com/10.1007/978-3-642-27311-7
[320] P. Sundberg, T. Brox, M. Maire, P. Arbelaez, and J. Malik, "Occlusion boundary detection and figure/ground assignment from optical flow," Computer Vision and ..., 2011. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs\{_\}all.jsp? arnumber=5995364
[321] J.-P. Tardif, "Non-iterative approach for fast and accurate vanishing point detection," 2009 IEEE 12th International Conference on Computer Vision, no. Iccv, pp. 1250-1257, 2009. [Online]. Available: http://ieeexplore.ieee.org/xpl/freeabs\{_\}all. jsp?arnumber=5459328
[322] W. M. B. Tiest and A. M. L. Kappers, "Tactile perception of thermal diffusivity." Attention, perception \& psychophysics, vol. 71, no. 3, pp. 481-489, 2009.
[323] R. Toldo and A. Fusiello, "with J-Linkage," Computer Vision - ECCV, pp. 537-547, 2008.
[324] X. Wu and X. Zhang, "Image interpolation using texture orientation map and kernel fisher discriminant," Image Processing, 2005. ICIP 2005. IEEE ..., pp. 9-12, 2005. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs\{_\}all.jsp?arnumber= 1529684
[325] B. Zhang, H. Zhu, and W. Zhang, "An efficient image simplification algorithm for brain MRI segmentation based on downhill filter," International Journal of numerical analysis and modeling, vol. 9, no. 2, pp. 378-384, 2012.
[326] L. Zhao and Y.-H. Yang, "Theoretical analysis of illumination in PCA-based vision systems," Pattern Recognition, vol. 32, no. 4, pp. 547-564, apr 1999. [Online].

Available: http://linkinghub.elsevier.com/retrieve/pii/S0031320398001198
[327] L. Zhao and Y.-h. Yang, "An efficient algorithm to compute eigenimages in PCAbased vision systems," Pattern Recognition, vol. 32, no. 5, pp. 851-864, may 1999. [Online]. Available: http://linkinghub.elsevier.com/retrieve/pii/S0031320398000326
[328] Y. Zhao, W. Gui, and Z. Chen, "Edge Detection Based on Multi-Structure Elements Morphology," 2006 6th World Congress on Intelligent Control and Automation, no. 4, pp. 9795-9798, 2006. [Online]. Available: http: //ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1713908
[329] Q. Zhu, G. Song, and J. Shi, "Untangling Cycles for Contour Grouping," 2007 IEEE 11th International Conference on Computer Vision, no. c, pp. 1-8, 2007. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm? arnumber=4408929
[330] O. A. Zuniga and R. M. Haralick, "Integrated Directional Derivative Gradient Operator," IEEE Transactions on Systems, Man, and Cybernetics, vol. 17, no. 3, pp. 508-517, 1987. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs\{_\}all.jsp?arnumber=4309068http: //ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=4309068


[^0]:    

[^1]:    

[^2]:    

[^3]:    

[^4]:    

[^5]:    

[^6]:    

[^7]:    (Miii) C School of Engineering । Biomedical Engineering Introduction

