

A Software Automation Framework For Image-Typeface Matching In Graphic Design

By

Taylor Javier Morris

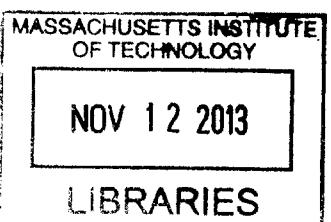
B.S. Mechanical Engineering
Louisiana State University—A&M, 2011

SUBMITTED TO THE DEPARTMENT OF MECHANICAL ENGINEERING IN
PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF

MASTER OF SCIENCE IN MECHANICAL ENGINEERING
AT THE
MASACHUSETTS INSTITUTE OF TECHNOLOGY

SEPTEMBER 2013

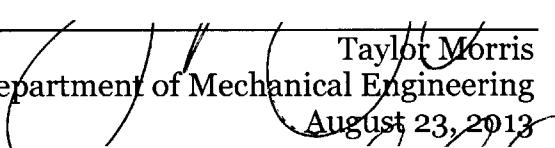
ARCHIVES



© 2013 Taylor J. Morris, All Rights Reserved

The author hereby grants to MIT permission to reproduce
and to distribute publicly paper and electronic
copies of this thesis document in whole or in part
in any medium now known or hereafter created.

Signature of Author:


Taylor Morris
Department of Mechanical Engineering
August 23, 2013

Certified by:


David Wallace
Professor of Mechanical Engineering, MacVicar Faculty Fellow
Thesis Supervisor, Extracurricular

Accepted by:


David Hart
Professor of Mechanical Engineering
Chairman, Committee for Graduate Students

This page intentionally left blank.

A Software Automation Framework For Image-Typeface Matching In Graphic Design

By

Taylor Javier Morris

Submitted to the Department Of Mechanical Engineering
on August 23, 2013 in Partial Fulfillment of the
Requirements for the Degree of Master of Science in
Mechanical Engineering

Abstract

This research proposes the framework for an automation tool that facilitates the graphic design process of image-font pairing or matching. Considering traditional graphic design principles, a multi-step software algorithm was developed to emulate the process of determining proportions and visual axes of both images and fonts. The algorithm then matches these visual markers using a decision hierarchy to produce a ranking of appropriate fonts from an existing font dataset.

To test the algorithm, 8 benchmark images were selected with varying proportions and visual axes. To build the font data set, each image was manually analyzed through a traditional graphic design process and then two fonts per image with similar, matching characteristics were manually selected.

The 8 benchmark images and 16 fonts were then used as inputs into the proposed matching software program. The results of the manually prescribed font-image pairings and calculated matches were then compared. Two images had the intended font in the top 4, two images had one of the intended fonts in the top 4, and 4 images had neither of the intended fonts in the top 4.

An additional step in image-font pairing includes detail matching by determining curvature similarities. This detail analysis will affect the pairing outcomes and should be further investigated. This research began to analyze these details, and makes recommendations for continuing this work. Additional future directions for this work include incorporating a user-interface to the matching algorithm, introducing expert testing, and down-selecting the first font pool based on deviation.

Thesis supervisor: David Wallace

Title: Professor of Mechanical Engineering, MacVicar Faculty Fellow

This page intentionally left blank.

Acknowledgements

To my friends and family...

“Feeling my way through the darkness,
Guided by a beating heart.
I can't tell where the journey will end,
But I know where to start.”
(Avicii & Blacc, 2013)

This page intentionally left blank.

Outline

Abstract.....	3
Acknowledgements	5
List of Figures	8
1. Introduction	9
1.1. Research Motivation	10
1.2. Literature Review.....	14
1.2.1. Image Matching.....	14
1.2.2. Parametric Font Design.....	15
1.2.3. Digitally-available Font Resources	19
1.3. Thesis Outline	20
2. Methodology	21
2.1. Image Analysis	22
2.2. Testing Procedure	30
3. Results.....	33
4. Conclusion	37
4.1. Discussion	38
4.2. Future Steps	41
4.2.1. Feature Matching	41
4.2.2. Emotional Associations	41
4.2.3. User-interface	42
References	43

List of Figures

Figure 1- Examples of highly recognized logo designs	11
Figure 2- Results comparison between manually and automated pairings.....	13
Figure 3- Rhythm in font and image pairing	15
Figure 4- Decomposition of the Times character “b”	16
Figure 5- Parameterizable elements in font generation	17
Figure 6- Character skeleton and distribution in font generation	18
Figure 7- Generated font from user-interface parametric design	18
Figure 8- Flowchart summarizing algorithm	21
Figure 9- Thresholding an image using the Otsu method.....	23
Figure 10- Centroid calculation for example image	25
Figure 11- Image proportions translated into bounding box	25
Figure 12- Graphic explaining image-boundary terminology	27
Figure 13- Corner point analysis	27
Figure 14- Finding corner points on contours	28
Figure 15- Curvature breakdown for single contour on cake image	29
Figure 16- Control set of fonts and visual markers.....	30
Figure 17- Comparison of manually prescribed and automated markers.....	34
Figure 18- Comparison of manually selected and automated font choices	35
Figure 19- Different interpretations of visual marker on the dots image	39

1 *Introduction*

Given the motivation for this work, this chapter provides review of relevant literature, explaining the graphic design principles of image-font pairing, exploring previous research conducted on parametric font design, and discussing current digitally-available font resources. The chapter concludes with an outline of the thesis.

1.1 Research Motivation

Visual packaging design plays an essential role in consumer decisions and the purchasing of physical products. Packaging develops brand identity by including a specific logo and color scheme, as well as a visual introduction to the product. As products' quality and function become only marginally different in a market (Reimann, Schilke, & Thomas, 2010), businesses are focusing product differentiation on less tangible features such as aesthetics (Brunner, Emery, & Hall, 2009). A study in 2004 showed that aesthetic design triggers positive responses in consumers, creating an immediate desire to own the product (Norman, 2004). Other studies have gone to show that consumers have a higher willingness to pay for a product that is presented in an aesthetically pleasing way (Bloch, Brunel, & Arnold, Individual differences in the centrality of visual product aesthetics: Concept and measurement, 2003), and possess an immediate sense of pride in owning aesthetically pleasing products (Bloch, Seeking the ideal form: Product design and consumer response, 1995). In looking to explain these positive triggers, a behavioral-neural correlation was studied through the use of functional magnetic resonance imaging (fMRIs). Consumer brain activity was study during the process of selecting various products. These products varied in aesthetic versus standardized designs, as well as unknown versus well-known brands. This study showed that unknown brands with aesthetic packaging design were chosen over well-known brands with standardized packaging, despite higher prices. The study offered an explanation for this correlation by noting activation of the striatum, the part of the brain associated with reward, while selecting aesthetic preference, judging, and making the final decision (Reimann,

Zaichkowsky, Neuhaus, Bender, & Weber, 2010). This study gives hope to new businesses entering into a market by explaining the importance of visual design on generating interest in a designed product.

Many large companies are willing to pay high prices for a logo and visual identity that appeals to an evolving market. In the last few years, Pepsi Co. paid an estimated \$1 million, the British Broadcasting Corporation (BBC) paid an estimated \$1.8 million, and British Petroleum (BP) paid an estimated \$211 million for new logos and branding packages to support a fresher visual identity (Stampler, 2013). In looking at the Pepsi brand redesign, the process took approximately 5 months.



Figure 1-Examples of highly recognized logo designs.

These logos represent corporate graphic design and professional pairing of images and fonts (Stampler, 2013).

While large companies can afford to hire staff or contract graphic designers to create logos, packaging, and consistent visual identities, new business often have limited resources. Further as software tools make the mechanics of graphic layout

accessible to the layperson, some may overlook the need for graphic design expertise. This renders these smaller companies unable to hire a graphics expert. The Kauffman Foundation, one of the largest foundations in the world devoted to entrepreneurship, estimated 514,000 new business owners each month during the year 2012 (Fairlie, 2013). While less-expensive crowd sourcing design services exist, such as 99designs.com which allows a person or company to “contract” a design job through a digital competition, the lowest tier of these services range between \$300-\$800 for a logo and up to \$1600 for a designed website. Additionally, the site advertises that paying a higher amount of money for a design attracts higher quality graphic designers. This implies that if a design is commissioned at a lower price-point, there is no guarantee that the design will be exceptional. As new businesses may not fully understand the importance of visual identity on consumer products, paying for such a service may not seem critical to launching a product or service.

To facilitate the logo and graphic design process, this work addresses a specific aspect of graphic design by introducing the framework for a tool that automates the process of matching images to fonts. This matching process is done by analyzing visual elements and with these elements, creating visual congruency between a selected image and font. Incongruence can be considered a form of information ambiguity (van Rompay, Pruyn, & Tieke, Symbolic Meaning Integration in Design and its Influence on Product and Brand Evaluation, 2009), and products high in congruency facilitate impression formation (Hekkert,

2006). Thus, this tool aims to identify and create visual congruency to strengthen and differentiate a brand message.

To automate this matching process between fonts and images, an algorithm analyzes proportions and graphic axes between a prescribed set of typefaces and images. The program outputs a ranking of recommended pairings, allowing the user to choose the final combination. To test this algorithm, 8 images were first manually analyzed and paired with fonts. The program then conducted the automated analysis of each image and paired the fonts similarly. Figure 2 gives an example of a comparison between the manual and automated pairings.



Figure 2-Results comparison between manually and automated pairings.
An example of the (a) visual markers with paired fonts compared to the (b) calculated markers and recommended pairings automation.

This work serves as the groundwork for a digitally-available, automated tool that could grow to include color schemes, layouts, and design packages. Furthermore, myfonts.com, a database of downloadable fonts, currently offers over 108,000 digital fonts; thus, in the future this tool could serve as an initial down-selection tool for experienced graphic designers. Additionally, this work could provide non-designers with an affordable solution to creating compelling and appropriate designs.

1.2 Literature Review

This section highlights prior research into the areas of font research, automated design research, and graphic design principles. From this literature review, it was concluded that a font-to-image matching tool currently does not exist.

Additionally, these resources helped to guide the work and potential future research directions.

1.2.1 Image Matching

Before creating a program to pair images to fonts, basic graphic design principles need to be articulated. These elements include proportions, visual axes, and similar shapes along with the principles of balance and rhythm.

In a symmetric composition, a line of symmetry can be drawn to divide the composition into even, balanced components. In asymmetric compositions, a visual axis analogously represents an axis of symmetry by creating an implied line about which visual components are balanced (Lupton & Cole Phillips, 2008).

Various forms of balance exist, including space, contrast, texture, position, and color. To create consistency and a consistent visual motion between graphic elements, similar visual axes should exist between an image and font pairing.

Rhythm in design is a repetition of elements to create a harmonious composition. In matching fonts to images, a designer attempts to create a rhythm between typographic and visual elements. This rhythm can be seen in proportions and

feature matching. Figure 3 shows an example of how proportions and similar shapes create a rhythm in image-font pairing.

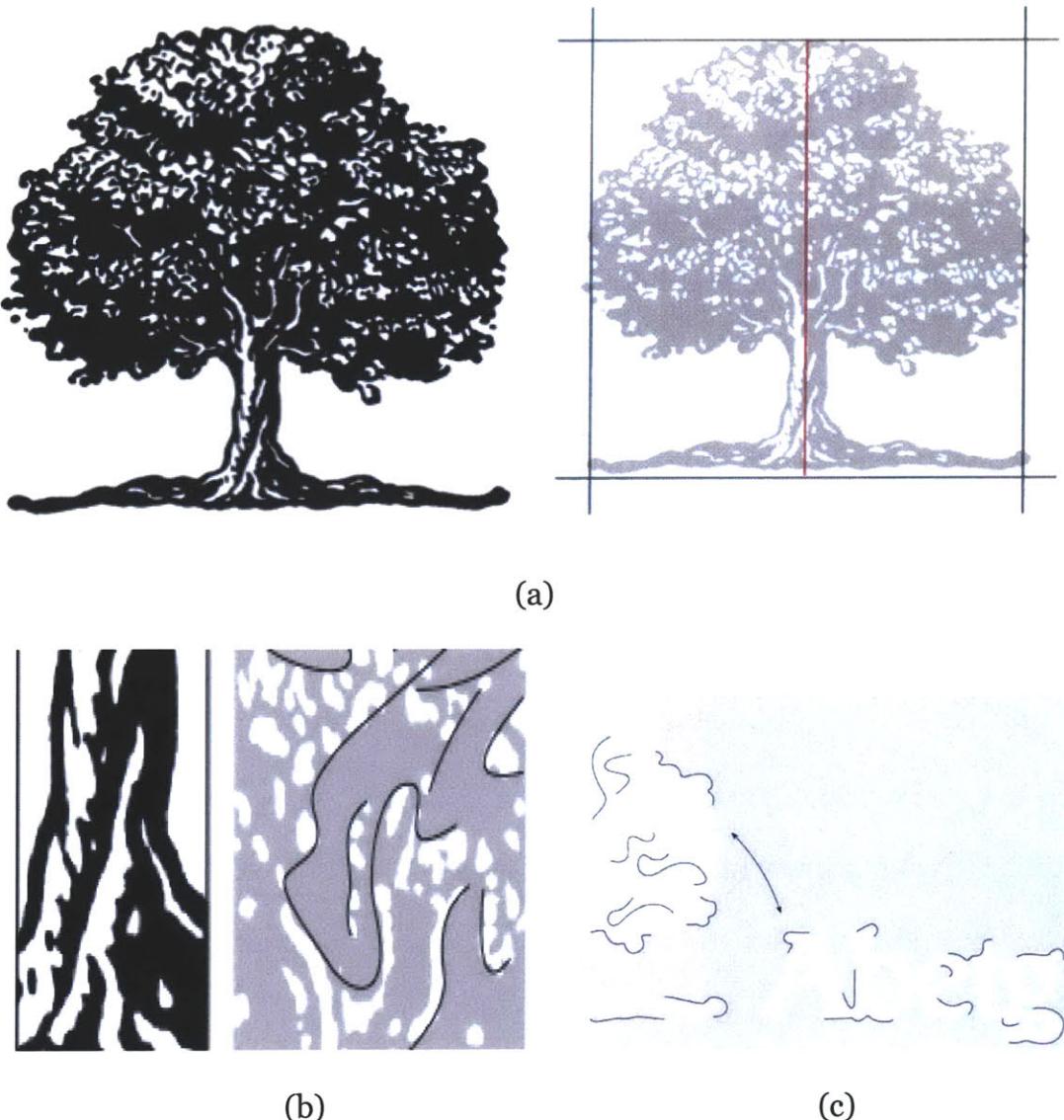


Figure 3-Rhythm in font and image pairing.

An example of how (a) proportions, (b) visual axes, and (c) shape features are analyzed between an image and font (Wallace, 2012).

1.2.2 Parametric Font Design

Between 1995 and 2001, Hu and Hersch developed a flexible way to define characters as a combination of parameterizable components. These parameters

included junction type, serif type, reference line position, bar and stem widths, and terminal metrics. Figure 4 shows a decomposition of a Times character using this method.

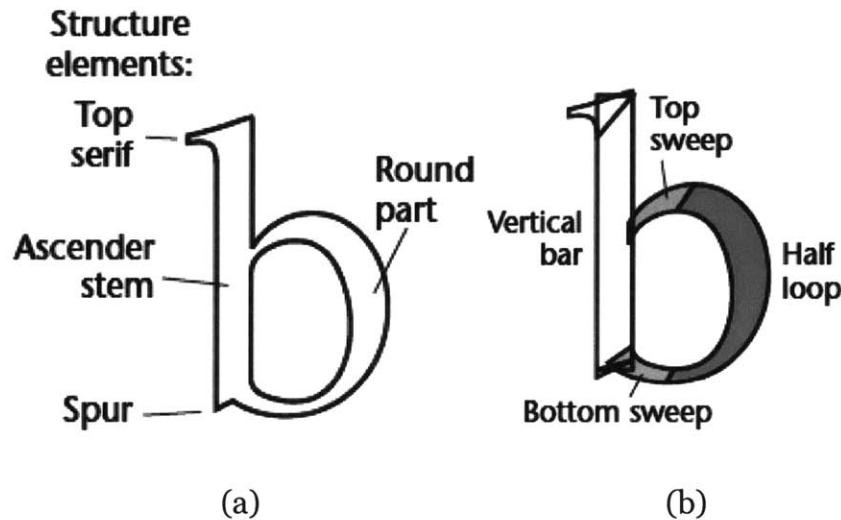


Figure 4-Decomposition of the Times character “b”.

The (a) structural components and (b) specific junction types of the character “b” (Hu & Hersch, 2001).

Once these components were identified and organized, each component could be modified to change line width, add or remove serifs, change corner treatments on characters, and change junction sweeps. Additionally, features such as bolding and italicizing could be easily added by modifying certain parameters. To smooth out the connection between elements, a curve-correcting Bezier was applied.

Figure 5 shows the synthesis of a character after parameterization.

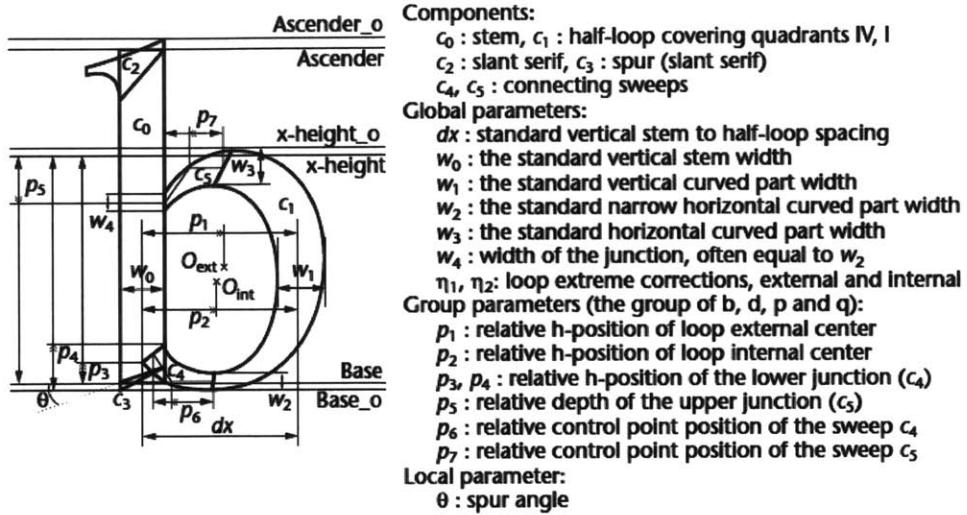


Figure 5-Parameterizable elements in font generation.

The synthesis of the Times character “b” with parameterized elements (Hu & Hersch, 2001).

Once the components were determined, they were applied across all characters.

The 26 lower-case characters in font Time required approximately 540 parameters (Hu & Hersch, 2001). In 2010, this method was revisited to extend the procedure to the Frutiger typeface. This study focused more specifically on adjusting boldness and exploring half-loop sweeps (Hassan, Hu, & Hersch, 2010).

Also in 2010, a different font-generation tool was developed based on user-interaction. The tool allows the user to draw a few characters, and then extracts the features to generate form and weights. The user first draws the outline; then the program computes a skeleton. The user can then adjust the skin, which is the decorative styling, and skeleton until a satisfactory result is presented. The skin is attached to the skeleton, so as the skeleton is adjusted, the skin adjusts accordingly. This feature preservation can be seen in Figure 6.

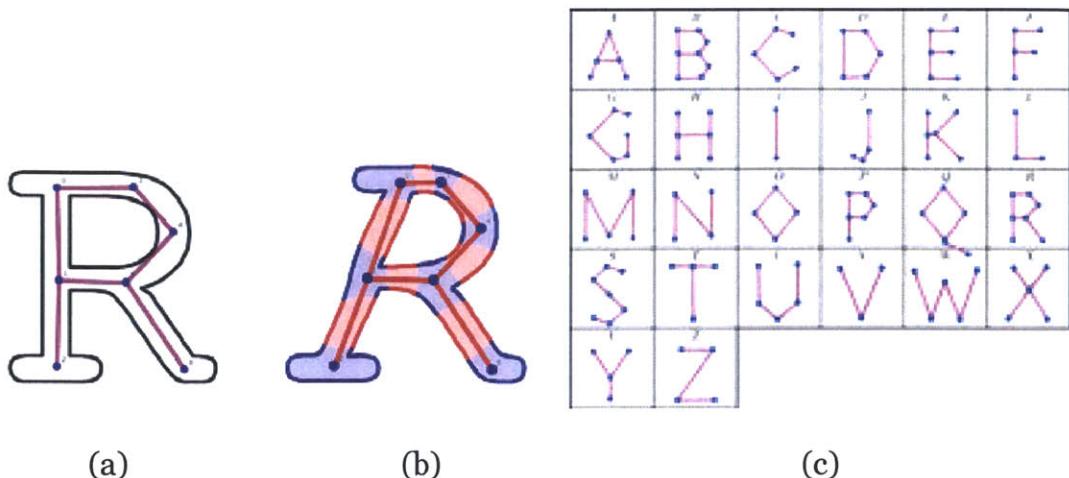


Figure 6-Character skeleton and distribution in font generation.

The skeleton is derived from the user input (a), and an appropriate, deformable skin (b) is attached. These features are then distributed to all characters (c) (Sueveranont & Igarashi, 2010).

Once the skin and skeleton of the character are confirmed, the program searches a database of fonts for similar looking characters. The program then pulls parameters from these fonts to create a blended font. The skeleton, skin, and blend then extend to the entire alphabet. Figure 7 compares the computed blended font with the drawn character and contributing fonts.

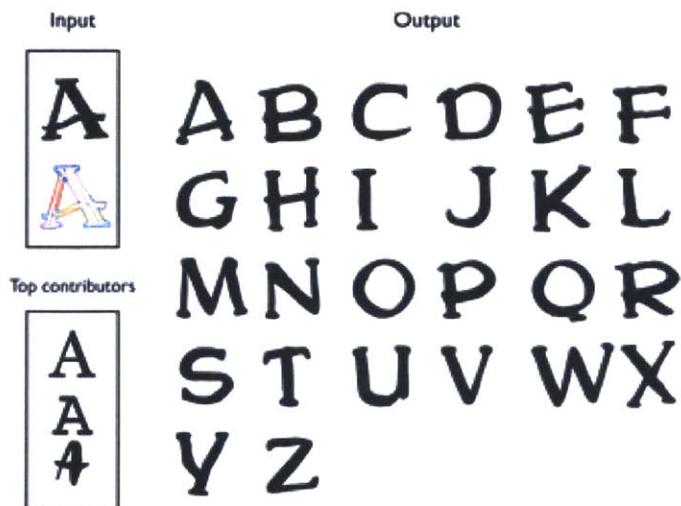


Figure 7-Generated font from user-interface parametric design.

Demonstration of how a set of characters was generated from a bend of an initial input and catalogued fonts (Suveeranont & Igarashi, 2010).

A limitation of this work included the inability to generate highly stylized fonts (Suveeranont & Igarashi, 2010).

1.2.3 Digitally-available Font Resources

108 fonts come preinstalled on a Mac with the OS X operating system. In addition, Myfonts.com offers over 108,000 digitally available fonts. These fonts are searchable through a wide array of categorizations, including webfont support and user tags (i.e. descriptive keywords applied to fonts by users to help group similar fonts). Dafont.com, another large database with approximately 21,000 free-to-download fonts, categorizes their fonts based on styles, such as holiday, techno, and foreign. Although these categories assist the designer in choosing a consistent theme between the type and the graphic message, font anatomy information is not available. Other notable digital font databases include fonts.com, fontsquirrel.com, and fontspring.com.

One font-oriented tool of interest is WhatTheFont, offered through myfonts.com (MyFonts, 1999). This tool allows a user to find the name of a font that they have seen in practice. The tool works by requesting an uploaded image of the font, preferably distinct characters of the font. The tool then performs a character matching analysis to find the font in the myfonts database. A WhatTheFont forum is implemented to supplement the tool, thus allowing a user to post a font and solicit feedback from other users on the name of the font or similar types of fonts.

Similar to the WhatTheFont tool, Identifont.com is a site that allows users to find fonts by font anatomy, compatibility between fonts, and similarity between fonts (Identifont, 2000). Although driven by a selection hierarchy instead of an algorithm, Identifont offers font identification as well as font pairings based on optical sizes, equal width fonts, and font companions. These offerings make Identifont a more practical tool for using different fonts in an applied setting, such as web layouts and print material, where multiple fonts may be utilized.

1.3 Thesis Outline

Following the introduction, Chapter 2, methodology , discusses how the algorithm automates the image-font pairing process. This chapter also includes the testing plan for comparing the manual pairings with the automated pairings. Chapter 03 presents the results of the work. In chapter 04, the final chapter of this thesis, the conclusion provides the results of the algorithm and a comparison between the manual and automated matches. A discussion of these results is complimented with potential explanations and future recommendations for addressing concerns such as a limited font sample size and subjectivity of the designer. The thesis concludes with future directions for the work.

2 Methodology

The algorithm starts by uploading an image. The proportions and visual axis of the image are then respectively determined through bounding box and best-fit curve calculations. Visual markers for a pool of fonts, which is determined through similar analysis, is compared with the calculated image markers. This comparison is first conducted with proportions to decrease the overall font pool size. The visual axis comparison then determines the final top 4 font pairings.

Figure 8 summarizes this process. The following section provides more detail into the specifics of the algorithm and testing procedure.

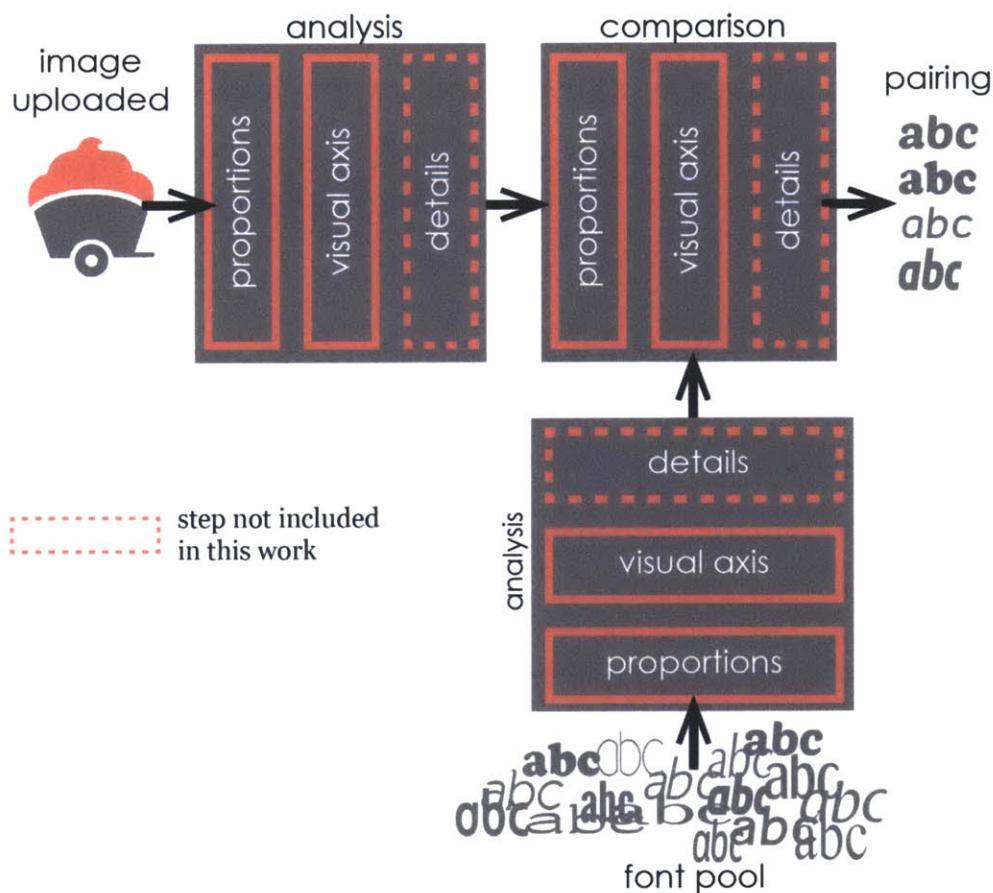


Figure 8-Flowchart summarizing algorithm.

This visual overview of the algorithm explains how an image is uploaded, analyzed, and compared based on a decision hierarchy.

2.1 Image Analysis

The algorithms used to match image to typeface is written in the Python programming language. In addition to traditional functions, three add-on libraries were used: SciPy, matplotlib, and scikit-image. Scikit-image is a collection of open-source, peer-reviewed algorithms for image processing. This package facilitated image manipulation and evaluation, such as thresholding the image, finding contours, and converting the image between formats. Scipy is an open source collection of scientific libraries and tools. The specific package in Scipy used in this work was numpy, a package for scientific computing. Scipy and numpy were used for matrix manipulation and curve-fitting. Matplotlib is a 2-dimensional plotting library for python. This library was used for transforming data into figures. Matplotlib is highly compatible with Scipy's data output formats.

The analysis begins by first opening and thresholding an image. The Otsu method is used to perform a histogram shape-based image thresholding. The algorithm separates an image into background and foreground, then calculates the optimum threshold separating these two classes so that their combined spread is minimal (Otsu, 1979). The outputted image is a binary sequence of filled and unfilled pixels. Figure 9 shows this process with the associated histogram selection.



Figure 9-Thresholding an image using the Otsu method.

This action converts the image to solid black based on a threshold value.

Each filled pixel is identified by its coordinate on an x-y axis drawn with origin at the upper left-hand corner of the image.

The x and y coordinates each have an array of values where the index of an element corresponds to the position on the axis. In this form, a filled pixel (i.e. a pixel that is read as black), contains a 1 in both the x and y index that corresponds to the coordinate of that pixel. For example, take a pixel located at (3,1) in the Cartesian coordinate system. In the format mentioned above, this point in x and y arrays is in the form of equation 1.

$$X = [0 \ 0 \ 0 \ 1 \ 0] \quad (\text{Equation 1})$$

$$Y = [0 \ 1 \ 0 \ 0 \ 0]$$

To convert this binary format to Cartesian coordinates, the indices of all 1 elements in these sparse arrays were indexed into a new matrix. These indices were then treated as coordinates. Corresponding to the above arrays, equation 2 shows the new format of these points.

$$X = [3] \quad (\text{Equation 2})$$

$$Y = [1]$$

$$XY = [(3,1)]$$

Now that the pixels can be interpreted as data points, the image analysis begins.

The first variable of comparison between image and type is proportions. To determine where the mass of the image was weighted, the centroid was calculated. To find the centroid, the arithmetic mean position of each point was calculated using equation 3 (Hibbeler, 2008).

$$\bar{x} = \frac{\sum_{i=1}^n x_i A_i}{\sum_{i=1}^n A_i} \quad (\text{Equation 3})$$

$$\bar{y} = \frac{\sum_{i=1}^n y_i A_i}{\sum_{i=1}^n A_i}$$

In equation 3, \bar{x} and \bar{y} are the x and y position of the centroid, n is the number of pixels, x_i and y_i are the centroidal position of a specific pixel, and A_i is the area of the corresponding pixel. Because each pixel has an identical area, the area term, A_i , factors out of the numerator and denominator, resulting in equation 4.

$$\bar{x} = \frac{\sum_{i=1}^n x_i}{n} \quad (\text{Equation 4})$$

$$\bar{y} = \frac{\sum_{i=1}^n y_i}{n}$$

Figure 10 illustrates this calculation.

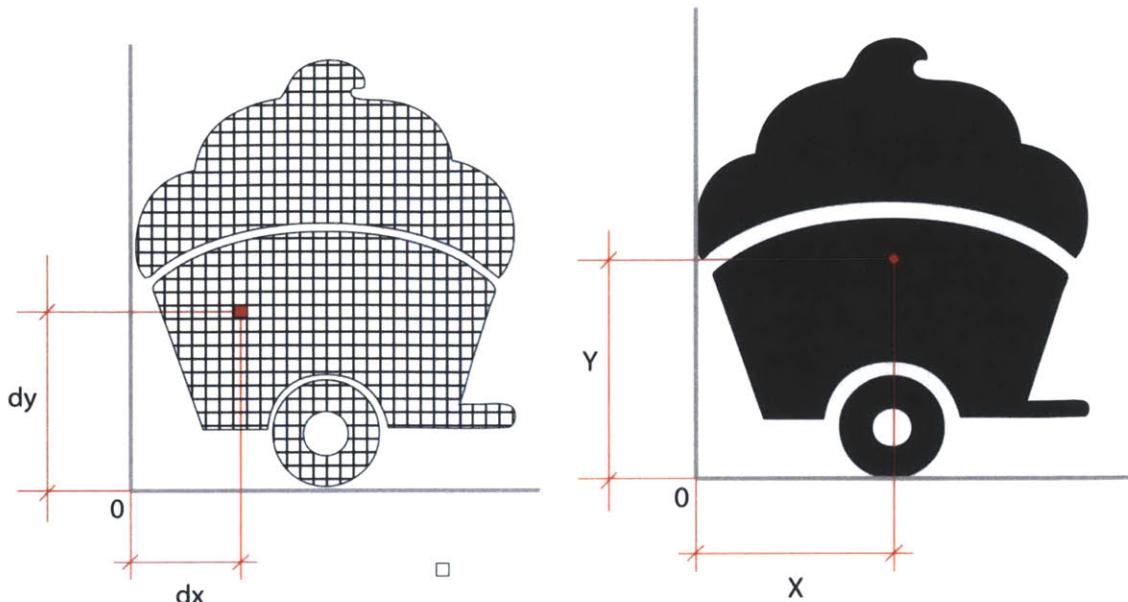


Figure 10-Centroid calculation for example image.
The centroid is calculated using the pixels as data points.

The standard deviation was then calculated from this centroid to eliminate outliers and to read a more concentrated mass, points outside of two standard deviations in both the x- and y-directions respectively were eliminated.

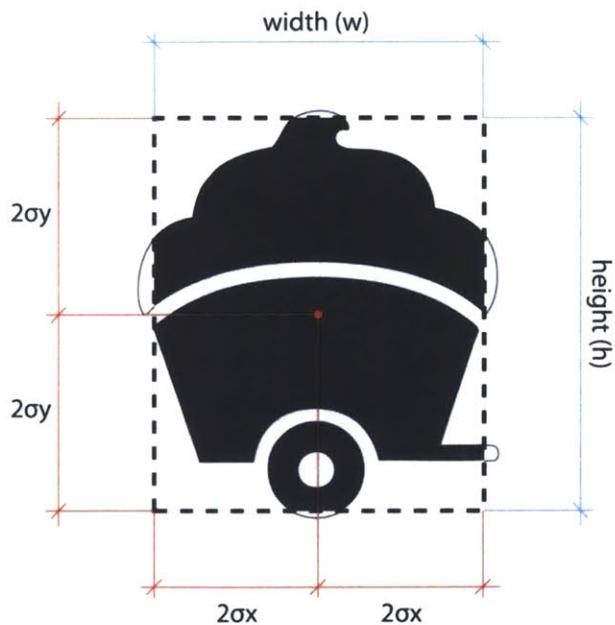


Figure 11-Image proportions translated into bounding box.

The bounding box represents the proportions of the image, calculated from the centroid.

The rectangle around the remaining pixels serves as a bounding box. The aspect ratio of this box represents the proportions of the image and was used for matching the image to a typeface.

The next variable to consider was the primary visual axis. Using the Cartesian coordinates of the image's pixels, an orthogonal regression, also known as total least squares, was used to plot a best-fit linear curve through the bounded pixels. An orthogonal regression minimizes the distance from an approximating line to a set of data points, which for this instance is a set of pixels (Stein, 1983). This regression method was chosen as an attempt to mathematically calculate a line of symmetry by minimizing pixel distances to the visual axis. The initial guess for this regression was a least squares linear regression. The slope of the approximated axis is then used as a comparison variable between image and font.

Analysis on the third variable, detail matching, was begun on this research. Detail extraction was completed for a prescribed image, however matching this variable should be investigated further. To better understand the terminology in detail extraction, the terms "contour", "curve", and "corner point" are explained in Figure 12 and defined as such. The contour refers to the outline drawn around filled pixels of the same value in an image. The contours of the image are calculated once the image is thresholded, thus pixels of the image are viewed as either "filled", (i.e. black) or "empty" (i.e. white). An image can possess multiple contours if the image has solid filled bodies separated by empty white space. These contours are then subdivided into "curves", which refers to a section of a

contour that possesses a beginning point, ending point, and peak. This curve analysis is one of the matching criteria used by the program.

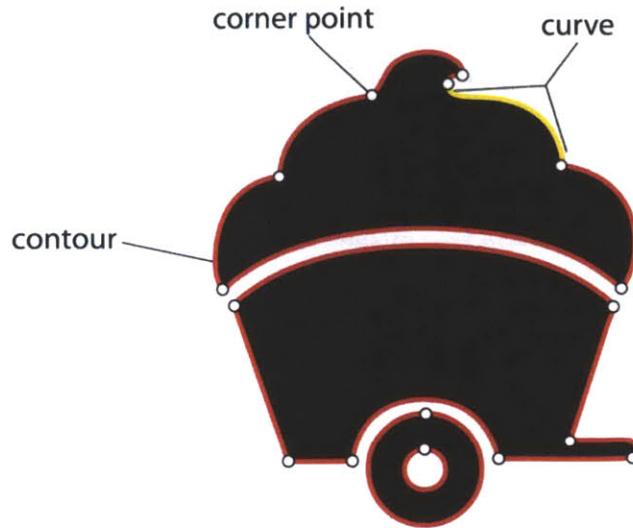


Figure 12- Graphic explaining image-boundary terminology.
This figure defines corner point, curve, and contour; three components used in the analysis.

The third factor for comparison is the corner treatment of the image and typeface characters. To perform this analysis, the contours of the image are found through a marching squares analysis. The Harris method is then used on the image to detect the corner points (Harris & Stephens, 1988).

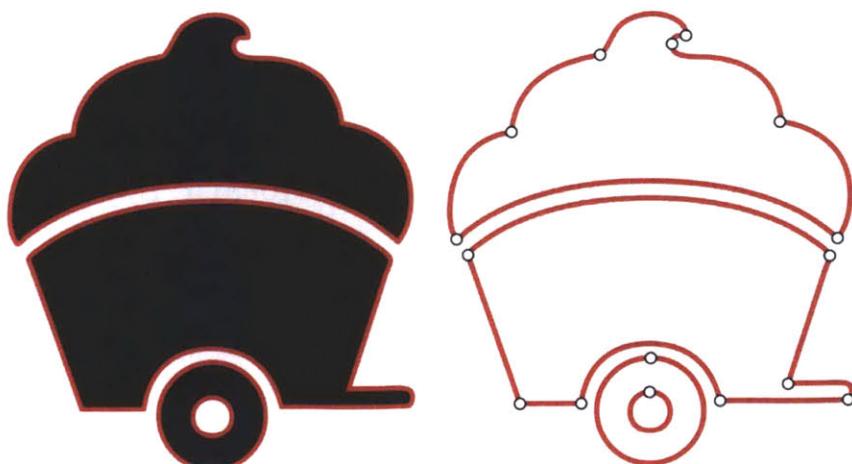


Figure 13-Corner point analysis.
Using the Harris method to detect the corner points.

Using a nearest neighbor approximation, the stored points were tested against the contour matrix to match corner points to their respective contour. This approximation minimized the distance between the corner point and nearest contour pixel. To calculate the minimized distance, D , equation 5 was used.

$$D = \sqrt{x^2 + y^2} \quad (\text{Equation 5})$$

Figure 14 explains this calculation.

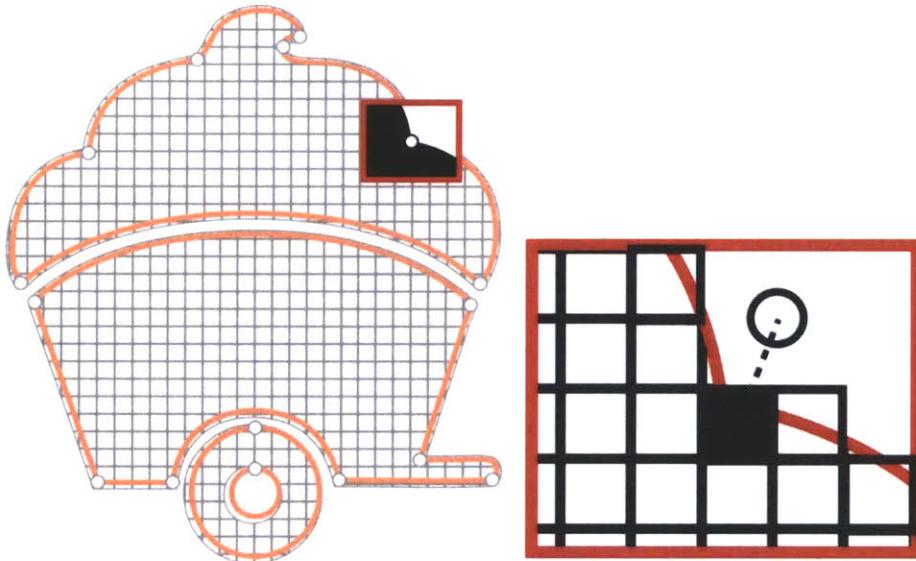


Figure 14-Finding corner points on contours.

The corner points were found on the contour using a nearest neighbor calculation.

Next each contour was analyzed independently. The contour data was organized such that each point was sequential to creating the contour. The corner points were then used as indices to determine the beginning and end points of a corner curve. The program first identified a corner coordinate on the contour, then identified the point before as the beginning point and point after as the end point.

All points between the beginning and end points were then added to a separate array for analysis.

This segment of the contour was then fitted with a cubic spline. Figure 16 shows how the curves of the cake image were divided and curve-fitted.

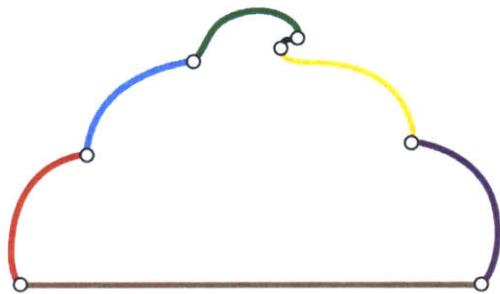


Figure 15-Curvature breakdown for single contour on cake image.
A demonstration of breaking a curve apart.

Next steps of this analysis would be to conduct a similar analysis on selected fonts.

In analyzing typefaces, the lowercase characters “o” and “l” were chosen for comparing proportions and visual axis respectively. These letters were chosen for their simplicity and relative standardization across fonts. To prepare the fonts for analysis, the characters were typed and exported as image files. The fonts were then uploaded to the program and analyzed in a similar format as the image. The program then made a comparison between the prescribed image and the stored data.

2.2 Testing Procedure

To test the algorithm, 8 images were chosen. These images represented variation in proportions and visual axes. Each of these images was paired with two fonts through the traditional graphic design process, as seen in Figure 16.

image markers	font markers	image markers	font markers
cake	poetsen one abc <input type="checkbox"/> <input checked="" type="checkbox"/> cooper black abc <input type="checkbox"/> <input checked="" type="checkbox"/>	flame	absolut pro abc <input type="checkbox"/> <input checked="" type="checkbox"/> vera humana 95 abc <input type="checkbox"/> <input checked="" type="checkbox"/>
cloud	tellural abc <input type="checkbox"/> <input checked="" type="checkbox"/> manksans abc <input type="checkbox"/> <input checked="" type="checkbox"/>	cone	league gothic abc <input type="checkbox"/> <input checked="" type="checkbox"/> designosaurus abc <input type="checkbox"/> <input checked="" type="checkbox"/>
dots	world of water abc <input type="checkbox"/> <input checked="" type="checkbox"/> round abc <input type="checkbox"/> <input checked="" type="checkbox"/>	arrow	dream orphanage abc <input type="checkbox"/> <input checked="" type="checkbox"/> doulos abc <input type="checkbox"/> <input checked="" type="checkbox"/>
coffee	luisa abc <input type="checkbox"/> <input checked="" type="checkbox"/> angleterre book abc <input type="checkbox"/> <input checked="" type="checkbox"/>	pencil	raspoutine classic abc <input type="checkbox"/> <input checked="" type="checkbox"/> natura play abc <input type="checkbox"/> <input checked="" type="checkbox"/>

Figure 16-Control set of fonts and visual markers.
Prescribed visual markers and recommended fonts for matching.

The images and fonts were analyzed using the software program for characterization. The program then used a decision hierarchy to perform an initial down-selection based on proportion matching. This down-selection narrowed the selection to eight fonts. A second down-selection was then conducted using the visual axis comparison. The program then output a rank order of fonts based on comparing the calculated values between font and image.

This page intentionally left blank.

3 Results

In looking at this first screening, 6 of the images had font lists that contained both intended fonts, while the other two font lists only contained one of the intended fonts. Two font lists, cone and flame, matched the top two fonts based on proportions, while the cloud image ranked the intended fonts first and third. Additionally, the coffee image ranked one of the intended fonts as first in its respective font list. In the second screening, 2 images had the intended font in the top 4, 2 images had one of the intended fonts in the top 4, and 4 images had neither of the intended fonts in the top 4. The cloud image correctly matched the first ranking font, Tellural, while the arrow and pencil images ranked one intended font in the top 2 choices. Figure 17 shows the correlation between personal and computer matching. Figure 18 summarizes the results of the matching. This list represents the fonts that first passed a proportions-screening, then ranked highest on visual axis.



Figure 17-Comparison of manually prescribed and automated markers.

Each image's prescribed visual markers were compared with the calculated markers.

manual image markers	manual font choices	calculated font choices
cake	<p>poetsen one abc <input type="checkbox"/> <input checked="" type="checkbox"/></p> <p>cooper black abc <input type="checkbox"/> <input checked="" type="checkbox"/></p>	<p>1) tellural <i>abc</i> <input type="checkbox"/> <input checked="" type="checkbox"/></p> <p>2) natura play <i>abc</i> <input type="checkbox"/> <input checked="" type="checkbox"/></p> <p>3) doulos <i>abc</i> <input type="checkbox"/> <input checked="" type="checkbox"/></p> <p>4) manksans <i>abc</i> <input type="checkbox"/> <input checked="" type="checkbox"/></p>
cloud	<p>tellural <i>abc</i> <input type="checkbox"/> <input checked="" type="checkbox"/></p> <p>manksans <i>abc</i> <input type="checkbox"/> <input checked="" type="checkbox"/></p>	<p>1) tellural <i>abc</i> <input type="checkbox"/> <input checked="" type="checkbox"/></p> <p>2) natura play <i>abc</i> <input type="checkbox"/> <input checked="" type="checkbox"/></p> <p>3) doulos <i>abc</i> <input type="checkbox"/> <input checked="" type="checkbox"/></p> <p>4) manksans <i>abc</i> <input type="checkbox"/> <input checked="" type="checkbox"/></p>
dots	<p>world of water <i>abc</i> <input type="checkbox"/> <input checked="" type="checkbox"/></p> <p>round <i>abc</i> <input type="checkbox"/> <input checked="" type="checkbox"/></p>	<p>1) tellural <i>abc</i> <input type="checkbox"/> <input checked="" type="checkbox"/></p> <p>2) natura play <i>abc</i> <input type="checkbox"/> <input checked="" type="checkbox"/></p> <p>3) dream orphan <i>abc</i> <input type="checkbox"/> <input checked="" type="checkbox"/></p> <p>4) doulos <i>abc</i> <input type="checkbox"/> <input checked="" type="checkbox"/></p>
coffee	<p>luisa <i>abc</i> <input type="checkbox"/> <input checked="" type="checkbox"/></p> <p>angleterre book <i>abc</i> <input type="checkbox"/> <input checked="" type="checkbox"/></p>	<p>1) tellural <i>abc</i> <input type="checkbox"/> <input checked="" type="checkbox"/></p> <p>2) natura play <i>abc</i> <input type="checkbox"/> <input checked="" type="checkbox"/></p> <p>3) manksans <i>abc</i> <input type="checkbox"/> <input checked="" type="checkbox"/></p> <p>4) raspoutine <i>abc</i> <input type="checkbox"/> <input checked="" type="checkbox"/></p>
flame	<p>absolut pro abc <input type="checkbox"/> <input checked="" type="checkbox"/></p> <p>vera humana 95 <i>abc</i> <input type="checkbox"/> <input checked="" type="checkbox"/></p>	<p>1) dream orphan <i>abc</i> <input type="checkbox"/> <input checked="" type="checkbox"/></p> <p>2) doulos <i>abc</i> <input type="checkbox"/> <input checked="" type="checkbox"/></p> <p>3) manksans <i>abc</i> <input type="checkbox"/> <input checked="" type="checkbox"/></p> <p>4) raspoutine <i>abc</i> <input type="checkbox"/> <input checked="" type="checkbox"/></p>

manual image markers	manual font choices	calculated font choices	
cone 	league gothic abc   designosaurus abc  	1) round dbc   3) doulos abc  	2) dream orphan abc   4) designosaurus abc  
arrow 	dream orphan abc   doulos abc  	1) round dbc   3) doulos abc  	2) dream orphan abc   4) designosaurus abc  
pencil 	raspoutine classic abc   natura play abc  	1) tellural abc   3) doulos abc  	2) natura play abc   4) manksans abc  

Figure 18-Comparison of manually selected and automated font choices.
Table of font recommendations produced from the matching program.

4 Conclusion

This work investigated the creation of an automated image-font pairing tool. Thus far, this thesis discussed the motivation for this work to assist both experienced and non-experienced designers in navigating large databases of fonts and making appropriate image-font pairings. The graphic design principles associated with image-font pairing were stated, as well as an investigation into parametric font design research. An overview of how the algorithm operated was presented, followed by a detailed explanation of each step of the algorithm. A manual analysis of image-font pairing was conducted on 8 benchmark images, each of which was paired with two appropriate fonts. The 8 benchmark images and 16 fonts were then used as inputs into the proposed matching software program. The results of the manually prescribed matches and calculated matches were then compared.

From this experiment, two images had the intended font in the top 4, two images had one of the intended fonts in the top 4, and 4 images had neither of the intended fonts in the top 4. The following sections discuss possible improvements and variations to the experiment, and provide potential future directions for the work.

4.1 Discussion

In looking at the results, three concerns should be addressed. The first observation was that of all images, the dots image had the least amount of correlation between the personal and computed visual markers. In Figure 17, it can be seen that the dots visual axis has an opposite sloping direction, and the calculated bounding box is more encompassing than the personally prescribed box. A possible explanation for this is that the image has a strong outlying shape (i.e. a seventh circle outside of the 6 circle array). This strong concentration of pixels shifts the distribution, thus affecting the centroid and bounding box calculations. As far as the visual axis, the image lends itself to multiple axes of symmetry. Because some dots are connected in the image, this creates visual movement to the viewer. The prescribed visual axis followed this movement, while the calculated axis appears to have focused more on symmetry.

In contrast to the dots image, the cloud image performed the best in matching a font. Unlike the dots image, the cloud image is a condensed shape. Thus, there is a more even distribution of pixels across the image, leading to a more accurate calculation of the bounding box. Although the calculated axis does not exactly match the prescribed axis, both axes represent a strong diagonal.

The second concern is that the prescribed visual markers are subjective to the designer's preference. To use the dots image as an example, the prescribed markers concentrated on the mass of the image. However, a different visual interpretation could be to match the font proportions based on a single dot in the

image. Figure 19 compares this possible interpretation with the prescribed markers. In attempt to compare the program to a professional designer, an expert visual designer with specialty in typography should be consulted to create visual markers. Additionally, the expert would rate the font sets' appropriateness for each image relative to the other fonts. In the larger sense, the program's effectiveness will also be evaluated based on how closely the automated pairing compares to the expert's pairing.

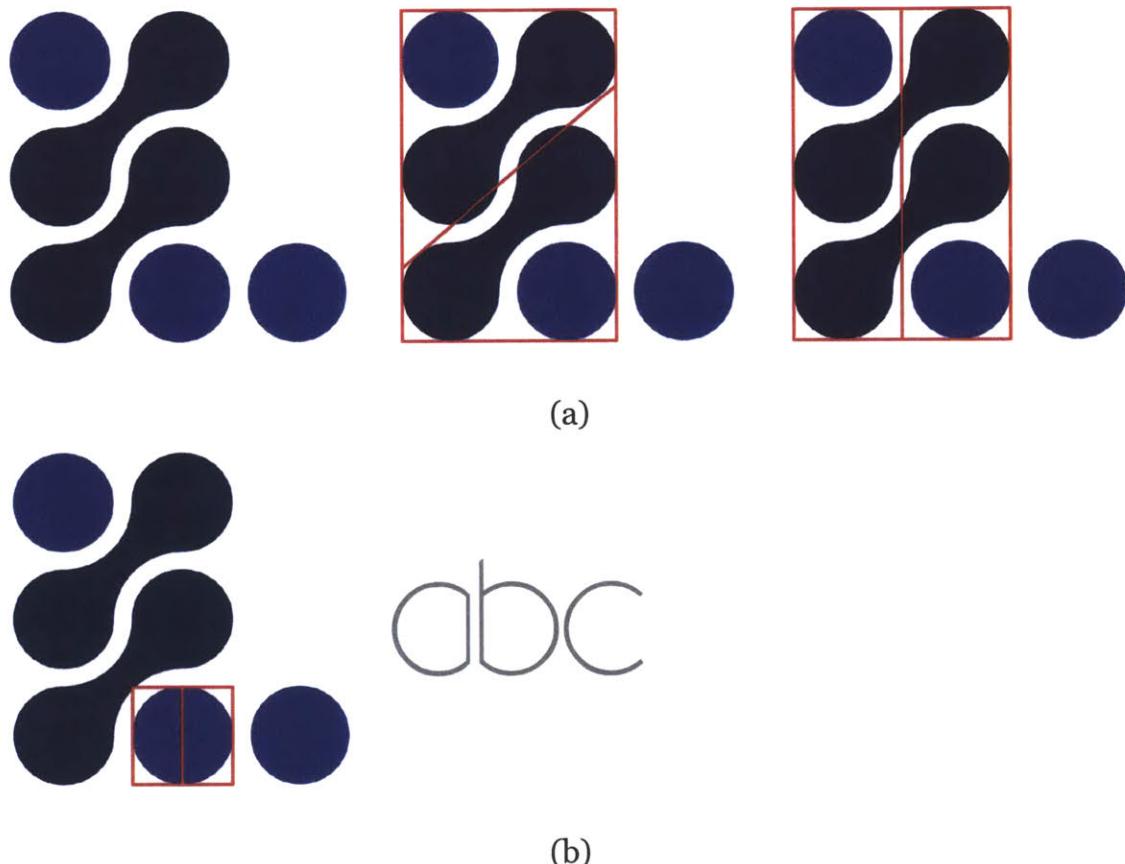


Figure 19-Different interpretations of visual marker on the dots image.
 (a) Different visual axes of the dots image, while (b) compares the same prescribed font, Round, to an individual object in the image.

The last point of concern is the sample size of the fonts available for pairing and the down-selection process. In practice, the pool of fonts would be extremely

large, thus there is a high potential for many fonts having similar proportions. The initial proportion-based down-selection process would not need to be limited to a set number, and the axis matching would be the deciding factor between fonts. In looking at this test, only 16 fonts were available for testing. Thus by forcing the program to select a subset of 8 fonts, there is the potential that some fonts in this initial list are not a good proportions match. A better approach might be to compare the proportions of the fonts with the image, and select fonts that are good proportions matches within a certain error. Each image may have a different size pool of fonts to choose from for visual axis matching, but this approach could provide more accurate pairings. Additionally, random, extreme examples could be added to the font pool as an experiment towards the algorithm's ability to discern non-related fonts.

4.2 Future Steps

In completing this research, several thoughts have arisen for potential future directions. These next steps include added features, added verification methods, and adjusted procedures.

4.2.1 Feature Matching

While this work tested proportions and visual axes, feature matching is another measure of visual congruency between images and fonts. The beginning of this analysis was conducted, as seen in section 2.1, and should be finalized to produce more appropriate image-font matchings. A similar ranking procedure should be conducted, with detail matching being the final decision in the decision hierarchy.

4.2.2 Emotional Associations

In early typographic studies, typography was categorized into different emotional associations. Font connotations have an impact on consumer perception of a product. Thus, a good image-font pairing should have both visual congruency and congruent meanings (Doyle & Bottomley, Dressed for the Occasion: Font-Product Congruity in the Perception of Logotype, 2006). Because a type's personality affects the verbal tone of the text (Brumberger, 2003), incorporating font connotations to the ranking process could improve the font ranking process. In relating these studies to this work, in addition to outputting a ranking of fonts, the program could output emotional associations or characteristics that the

proposed fonts have. In looking at myfonts.com, each font has a list of associated tags. These tags could be recommended by the creator of the font, but also added and modified by users of the font. This added benefit would allow the user to choose a font that has high visually congruent design elements as well as emotional appropriateness for the product.

4.2.3 User-interface

Incorporating more user interaction makes the use of the program more personal. While most of the matching process is automated, having the user's input for certain variables could increase the accuracy of the pairing algorithm. One proposed method for adding user interaction is to create a web interface for the program. The program could provide the user with an initial recommendation of proportions and visual axis. The user can then verify that these markers are accurate, with the ability to manipulate the markers as seen fit. Based on these verified markers, the program would match the image to fonts.

References

1. Arnell Group. (2008). *BREATHTAKING: Design Strategy*. Arnell Group.
2. Bloch, P. (1995). Seeking the ideal form: Product design and consumer response. *Journal of Marketing*, 16-29.
3. Bloch, P., Brunel, F., & Arnold, T. (2003). Individual differences in the centrality of visual product aesthetics: Concept and measurement. *Journal of Consumer Research*, 551-565.
4. Brumberger, E. (2003). The Rhetoric of Typography: The Awareness and Impact of Typeface Appropriateness. *Technical Communication*, 50, 224-231.
5. Brunner, R., Emery, S., & Hall, R. (2009). *Do you matter? How great design will make people love your company*. Upper Saddle River, New Jersey: FT Press.
6. Childers, T., & Jass, J. (2002). All Dressed Up With Something to Say: Effects of Typeface Semantic Associations on Brand Perception and Consumer Memory. *Journal of Consumer Psychology*, 93-106.
7. Doyle, J., & Bottomley, P. (2006). Dressed for the Occasion: Font-Product Congruity in the Perception of Logotype. *Journal of Consumer Psychology*, 112-123.
8. Doyle, J., & Bottomley, P. (2004). Font appropriateness and brand choice. *Journal of Business Research*, 873-880.
9. Doyle, J., & Bottomley, P. (2009). The Message in the Medium: Transfer of Connotative Meaning from Typeface to Names and Products. *Applied Cognitive Psychology*, 396-409.
10. Fairlie, R. W. (2013). *Kauffman Index of Entrepreneurial Activity*. Kansas City: Ewing Marion Kauffman Foundation.
11. Fennis, B., & Pruyn, A. (2006). You are what you wear: Brand personality influences on consumer impression formation. *Journal of Business Research*.
12. Harris, C., & Stephens, M. (1988). A combined corner and edge detector. *Proceedings of the 4th Alvey Visual Conference* (pp. 147-152). United Kingdom: The Plessey Company.
13. Hassan, T., Hu, C., & Hersch, R. (2010). Next Generation Typeface Representation: Revisiting Parametric Fonts. *10th ACM Symposium on Documenting Engineering* (pp. 181-184). Manchester: ACM.
14. Hekkert, P. (2006). Design aesthetics: Principles of pleasure in design. *Psychology Science*, 157-172.
15. Henderson, P., Giese, J., & Cote, J. (2004). Management Using Typeface Design. *Journal of Marketing*, 60-72.
16. Hibbeler, R. (2008). *Engineering Mechanics: Statics*. Upper Saddle River, New Jersey: Prentice Hall.
17. Hu, C., & Hersch, R. (2001). Parameterizable Fonts Based on Shape Components. *IEEE Software*, 70-85.
18. Identifont. (2000, November). *Identifont- Identify fonts by appearance, find fonts by name*. Retrieved August 20, 2013, from Identifont: www.identifont.com/

19. Lupton, E., & Cole Phillips, J. (2008). *Graphic Design The New Basics*. New York: Princeton Architectural Press.
20. Meggs, P. (1992). *Type and Image: The Language of Graphic Design*. New York, New York: John Wiley & Sons, Inc.
21. Mikaere, E. J. (2011). *How Favourable Attitudes are Formed when the Semantic Associations of a Logotype are Congruent with Brand Personality*. The University of Waikato, Department of Computer Science. Hamilton: Research Commons at the University of Waikato.
22. Mullet, K., & Sano, D. (1994). *Designing Visual Interfaces: Communication Oriented Techniques* (1st Edition ed.). Upper Sadle River, New Jersey, United States of America: Prentice Hall.
23. MyFonts. (1999). *WhatTheFont! << MyFonts*. Retrieved August 20, 2013, from MyFonts: The World's Largest Collection of Fonts: <http://www.myfonts.com/WhatTheFont/>
24. Norman, D. (2004). *Emotional design: Why we love (or hate) everyday things*. New York, New York: Basic Books.
25. Otsu, N. (1979, January). "A threshold selection method from gray-level histograms. *Systems, Man and Cybernetics, IEEE Transactions on* , 62-66.
26. Qing, H., Kai, Z., Zhang, C.-F., & Chen, M.-R. (2012). Packaging Design Research and Analysis Based on Graphic Visual. *IACSIT Coimbatore Conferences*. 28, pp. 148-153. Singapore: IACSIT Press.
27. Reimann, M., Schilke, O., & Thomas, J. (2010). Toward an understanding of industry commoditization: Its nature and role in evolving marketing competition. *International Journal of Research in Marketing* , 188-197.
28. Reimann, M., Zaichkowsky, J., Neuhaus, C., Bender, T., & Weber, B. (2010). Aesthetic package design: A behavioral, neural, and psychological investigation. *Journal of Consumer Psychology* , 431-441.
29. Stampler, L. (2013, March 26). *Here's How Much The World's Most Iconic Logos Cost Companies to Design Them*. Retrieved August 13, 2013, from Business Insider: <http://www.businessinsider.com/heres-how-much-the-worlds-most-iconic-logos-cost-companies-2013-3?op=1>
30. Stein, Y. (1983). Two Dimensional Euclidean Regression. *Conference on Computer Mapping*. Herzlia.
31. Suveeranont, R., & Igarashi, T. (2010). Example-Based Automatic Font Generation. In R. T. al., *Smart Graphics* (pp. 127-138). Berlin: Springer Berlin Heidelberg.
32. van Rompay, T., & Pruyn, A. (2011). When Visual Product Features Speak the Same Language: Effects of Shape-Typeface Congruence on Brand Perception and Price Expectations. *Journal of Product Innovation Management* , 599-610.
33. van Rompay, T., Pruyn, A., & Tieke, P. (2009). Symbolic Meaning Integration in Design and its Influence on Product and Brand Evaluation. *International Journal of Design* , 19-26.
34. Wallace, D. (2012, April 26). *The Messenger of Experience: Form and Color*. Retrieved August 13, 2013, from 2.744 Notes and Media: https://web.mit.edu/2.744/www/resourceMaterials/classNotesProtected/16_graphics.pdf