UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
INSTITUTO DE INFORMÁTICA
PROGRAMA DE PÓS-GRADUAÇÃO EM COMPUTAÇÃO

DENNIS GIOVANI BALREIRA

Efficient acquisition and synthesis in computerized handwriting

Thesis presented in partial fulfillment
of the requirements for the degree of
Doctor of Computer Science

Advisor: Prof. Dr. Marcelo Walter

Porto Alegre
December 2019

CIP – CATALOGING-IN-PUBLICATION

Balreira, Dennis Giovani

Efficient acquisition and synthesis in computerized handwriting / Dennis Giovani Balreira. – Porto Alegre: PPGC da UFRGS, 2019.

110 f.: il.

Thesis (Ph.D.) – Universidade Federal do Rio Grande do Sul. Programa de Pós-Graduação em Computação, Porto Alegre, BR–RS, 2019. Advisor: Marcelo Walter.

1. Computer graphics. 2. Image processing. 3. Handwriting synthesis. 4. Handwriting acquisition. 5. Character comparison. I. Walter, Marcelo. II. Título.

ACKNOWLEDGEMENTS

First I would like to dedicate a special thanks to my advisor Professor Marcelo Walter, for supporting and believing in me since the undergraduate course. I have learned so much in this period, mostly due to his guidance, patience and advice.

I would like to thank the Postgraduate Program in Computing (PPGC) for the opportunity and the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES) for the financial support.

I would like to acknowledge the Federal University of Rio Grande do Sul (UFRGS) and the Institute of Informatics, extending it to the professors and admin staff for all their daily effort to construct this great environment.

I would like to thank my family, specially my mother Maria José, for all the support along this academic journey.

I would like to thank my friends for the amazing support and my colleagues of the Institute of Informatics for the academic support.

ABSTRACT

Although nowadays people rely almost full time on digital text, the use of handwriting has earned a special status for specific cases. We still learn to write by hand and use it as an identifying tool throughout our entire life. The computerized handwriting area addresses solutions in the three main handwritten fields: acquisition, recognition, and synthesis. In particular, handwriting synthesis generates renderings of text which resemble natural handwriting but are, in fact, synthesized by a model. The results in this area have several uses, such as artistic applications, CAPTCHA generation, and by providing new examples for handwriting recognition. It is still a challenging research area since it is challenging to mimic natural handwriting due to individual characteristics. Most of the current research in the field present robust approaches with sophisticated techniques, such as neural networks, which require a large number of samples and deal with large time-consuming optimization problems. Although they produce natural-looking results overall, these techniques require a large number of resources, which most of the time make them non-trivial for daily usage. In this thesis, we introduce two main contributions to computerized handwriting. First, in the acquisition field, we present an in-depth look investigating the number of samples needed for reproducing natural variability. We show that writing samples collected from a minimal set are statistically equivalent in variation with larger sets. This discovery provides many benefits, such as the shorter time needed to collect the samples. Our samples were collected from a special tablet device that captures the users handwriting using pen and paper, without introducing typical distortion produced by graphics tablets. Second, in the synthesis area, we introduce a novel technique to generate handwriting from public fonts. Given a digitalized input sample of the desired handwriting, we present an algorithm that finds the best match between characters using as a source for the output text the extensive collection of publicly available fonts designed to look like handwriting. Our results show that even though human calligraphy is highly individual and specialized, visually similar renderings are possible for many applications that do not demand full similarity, considerably increasing its synthesis variability from a few inputs without the complexity of state-of-the-art approaches.
Keywords: Computer graphics. image processing. handwriting synthesis. handwriting acquisition. character comparison.

Aquisição e síntese eficientes em caligrafia computadorizada

RESUMO

Apesar de hoje em dia ser comum trabalhar durante quase todo tempo com textos escritos digitalmente, escrever textos à mão possui um *status* especial em casos específicos. Ainda é comum que pessoas aprendam a escrever à mão desde pequenas, inclusive sendo uma das formas mais conhecidas de identificação pessoal ao longo da vida. A área de escrita à mão computadorizada provê soluções nos seus três campos principais: aquisição, reconhecimento e síntese. Em particular, a área de síntese de caligrafia é responsável por gerar textos utilizando modelos matemáticos de forma que pareçam ter sido feitos por alguém à mão. Esta área possui diversos usos, sendo eles artísticos, na produção de novos *CAPTCHAS* e até mesmo servindo como novas entradas para o reconhecimento de escrita à mão. Ainda é um campo desafiador, pois é muito difícil imitar uma escrita à mão devido a características individuais. A maioria dos trabalhos atuais dessa área apresentam abordagens com técnicas complexas, como redes neurais, as quais necessitam de um grande número de amostras e lidam com longos problemas de otimização. Embora estas técnicas gerem bons resultados, também requerem uma grande quantidade de recursos, os quais na maioria das vezes se tornam não triviais para uso cotidiano. Nesta tese são exploradas novas abordagens para caligrafia computadorizada. Primeiramente, na área de aquisição, é apresentada uma investigação sobre o número de amostras necessárias para reproduzir variabilidade natural. É apresentado que amostras coletadas de um conjunto mínimo são estatisticamente equivalentes em variação quando comparadas a conjuntos maiores. Esta descoberta possui diversos benefícios, tais como tempo menor para coletar as amostras. Além disso, foi utilizado um dispositivo especial que captura a escrita à mão de usuários utilizando papel e caneta comum sem introduzir distorção dos *tablets*. Na área de síntese é apresentada uma nova técnica para gerar escrita à mão a partir de fontes públicas. Dado um texto de entrada digitalizado, é apresentado um algoritmo que encontra a forma mais similar entre diferentes caracteres utilizando para saída de dados uma grande coleção de fontes disponíveis publicamente. Nossos resultados mostram que, apesar da caligrafia humana ser altamente individual e específica, é possível obter bons resultados com baixo custo computacional, sendo útil para aplicações que não necessitam de alta similaridade.

Palavras-chave: computação gráfica, processamento de imagens, síntese de caligrafia, aquisição de caligrafia, comparação de caracteres.

LIST OF FIGURES

LIST OF TABLES

CONTENTS

1 INTRODUCTION

With the advance of digital technologies, many forms of communication became old-fashioned and were replaced by faster and more efficient ways, such as handwritten text. It is common sense that today we rely almost full time on typed text. However, handwriting messages are distinguished with a special status. They have uses in specific cases, suggesting a friendship between sender and receiver. In the industry, there is a specific market for letters written by hand, where companies handwrite the letter for customers (THE..., 2017). Figure 1.1 presents an example of a letter from this company. The opening sequence in the movie *Her* (HER, 2013) shows the main character in his job, dictating a love letter which is being rendered in real-time on the computer screen as written text from a particular customer. The final letter is printed and sent to the recipient as if it was written by the customer herself. The fictitious company is called <handwrittenletters.com>. In this not so distant future, such a company would need, as input for rendering the letter, a sample of the actual handwriting of its customers. In this context, algorithms that automatically generate handwriting have a special importance. After all, digital messages cost little to produce whereas handwritten text requires attention and more time than digital communications overall.

The handwriting area can be subdivided into three main categories: acquisition, recognition, and synthesis. *Acquisition* refers to how the handwriting data will be captured and stored. *Recognition* relates to identifying handwriting text already written by an user. Finally, *Synthesis* intends to generate new handwriting based on handwriting inputs. Although the acquisition is not a very important field alone, it is an indispensable step for the other two, since both need to work with user data. This work will focus on synthesis and acquisition, although our results can also be used for recognition.

The goal of the handwriting synthesis area is to provide images that look like they were written by a human hand. The research on this area started back in the early 90s (RAO, 1993) but it still attracts interest today (Graves, 2013) (HAINES; AODHA; BROSTOW, 2016). Synthesis can also be seen in general as the reverse of handwriting recognition. For instance, producing individual characters is the inverse of character recognition and generating words by concatenating the characters is the inverse of character segmentation.

One of the most important subdivision of the synthesis field is the distinction between offline or online data inputs. Offline data synthesis algorithms work with text with-

Figure 1.1 Example of a handwritten letter created by The Indian Handwritten Letter Co.

Source: The Indian Handwritten Letter Co. (THE. . . , 2017).

out temporal information, i.e., usually scanned or printed text images. In contrast, online data contains text with temporal information, i.e., mostly gathered with tablet-like devices as a set of points along time. Further details of this classification are discussed in Chapter 2. Current state-of-the-art synthesis techniques present robust approaches for both online and offline data. Some focus on neural networks (Graves, 2013), which requires a large amount of data, while others use advanced optimization techniques to select the best characters the user has written (HAINES; AODHA; BROSTOW, 2016). Although these techniques indeed produce natural-looking results, they require a large amount of resources and sampling.

Furthermore, current state-of-the-art handwriting synthesis techniques require the subject to spend around 40min writing the samples (HAINES; AODHA; BROSTOW, 2016). The main motivation for such long capture time is the need to capture variations in the real handwriting such that the synthesis can later use this variation. However, the current amount of input samples is defined in a conservative approach, that is, without actually answering the question of how many samples are needed to reproduce natural variability. An alternative to collect long texts is to collect pangrams, which are sentences that use every letter of the alphabet at least once. An example of a famous pangram in

English is *The quick brown fox jumps over the lazy dog*, containing 35 unique letters. The argument against pangrams is that they do not capture enough variation about a subject's handwriting. Therefore some techniques approach this by asking the user to write a reasonable amount of text to collect variability, as in (HAINES; AODHA; BROSTOW, 2016).

1.1 Applications

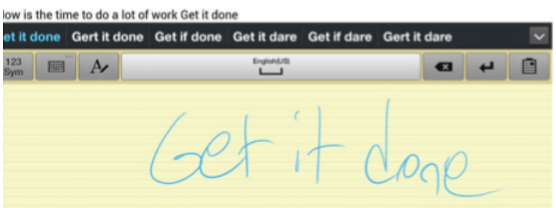
There are several applications which could benefit directly from handwritten synthesis algorithms. We detail some of the most important applications below. We exemplify the most relevant ones in Figure 1.2:

- **Improvement of Text Recognition Systems**: provides synthesized data for handwriting recognition algorithms. The process of producing new data, which is called data augmentation, is supposed to be easier than collecting it from users. Plus, handwriting recognition is a very important topic that is active nowadays because of the deep learning usage increase.
- **CAPTCHA generation**: newly synthesized examples can generate a large combination of letters to set in for Completely Automated Public Turing test to tell Computers and Humans Apart (CAPTCHA). By using non-user specific handwriting, for instance, it is possible to synthesize new kinds of letters and even mix letters from distinct users to become less readable.
- **User-specific Font Personalization**: even though nowadays we rely basically upon typed digital text, handwritten messages have a special place for particular uses. One may want to send an e-mail with its own calligraphy, for instance. Instead of writing the message on a sheet of paper by hand, scan and upload, it would be easier to just have a user personalized font which can be imported and digitally typed.

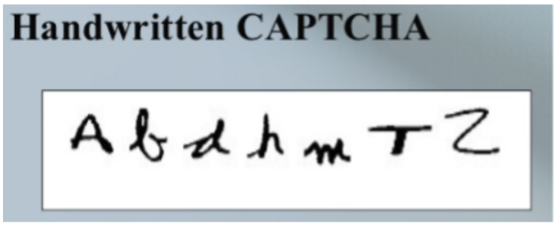
According to the survey of Elarian and colleagues (ELARIAN et al., 2014), the applications related to recognition are the most predominant in the literature, which can be justified due to its relevance.

Figure 1.2 Example of different applications for handwriting synthesis: (a) text recognition, (b) CAPTCHA and (c) artistic.

(a) Text recognition.



(b) CAPTCHA.



(c) Artistic.

Source: the author.

1.2 Goals and contributions

The goal of this thesis is to explore new approaches in the field of handwriting synthesis and acquisition. Our techniques focus on minimizing the input samples handwritten by the users and decrease the computational cost when compared with state-of-the-art techniques. Our work presents the following contributions: (i) a study which states that samples collected from a minimal set are statistically equivalent in variation with larger sets; (ii) a novel offline handwriting synthesis technique; and (iii) a new dataset obtained from a special tablet device that captures the user's handwriting with common pen and paper. The synthesis approach (ii) decreases the computational cost and is well suited for a low variability data sampling when compared to other state-of-the-art techniques. Our findings in our handwriting study (i) and data collection (iii) directly contribute for the handwriting area in general.

1.3 Overview

The rest of this thesis is organized as follows: Chapter 2 introduces the necessary background to this thesis, including the definition of handwriting specific vocabulary and other relevant terms used in our study and synthesis methods. Next, Chapter 3 presents the related work, covering the most relevant research for both online and offline handwriting synthesis area, plus general shape comparison methods and handwriting databases. We show our study about handwriting variability and present our dataset in Chapter 4 while Chapter 5 shows our handwriting synthesis approach for the offline data type. Finally, Chapter 6 presents our conclusions, limitations and future work.

2 BACKGROUND

In this chapter, we discuss the most relevant concepts of handwriting synthesis and further related information we will use in the next chapters. The recent survey by Elarian and colleagues (ELARIAN et al., 2014) provides a useful starting point. Besides covering previous research, they also organize very important concepts of the field. At the highest level, the methods are classified as either top-down or bottom-up. In the top-down approach, the solution tries to simulate the actual neuromuscular movements of the human arm and hand to produce the results. Bottom-up approaches, also known as shape simulation, model the shape of individual units to obtain the result. The aspects of handwriting synthesis can be classified as:

- **Level of granularity**: refers to the kind of output that is generated by the system. They can be presented as strokes, characters, words, lines, paragraphs, etc. Each approach can work with one or many levels of granularity, usually depending on the application.
- **Techniques**: can be classified into two types, generation and concatenation. In the generation, new instances are synthesized for a given writing unit. Concatenation connects smaller scripting units into larger ones.
- **Data types**: the data can be either online or offline. Online synthesis works with input text data containing temporal information. All x and y coordinates gathered from the capture device are stored according to the writing time sequence. Generally, online data are captured on tablet-like devices since they provide a just-in-time way of storing the strokes along with time. The devices may allow capturing coordinate time-stamps and pressure. On the other hand, offline synthesis deals with input text data without temporal information. In this case, a text is already written and there is no past record available. Offline data are usually stored as static images of a script written on paper, thus containing inking and thickness information. In general, it is desirable to work with online data since it contains more information to process than the static offline data. Still, there are cases where it is not possible nor desirable to obtain online handwriting, making offline data approaches a necessity. Techniques can be either one of them or both.
- **Scripts and Languages**: scripts are sets of characters, including signs or symbols, which are used for communication via reading and writing. Scripts can be cursive (Arabic), discrete (Hiragana) or mixed (modern Latin). One script can be used

to write different languages. There are many synthesis approaches covering one specific script or even more than one at the same time.

- **Writing variability**: can be either writer-specific or writer-independent, depending on the aimed application. For applications like user-specific font personalization, the writing variability must be concentrated on a single user. However, if the method intends to work with CAPTCHA generation or improvement of text recognition systems, it does not necessarily need to be focused on a writer in particular. There are systems which work for both solutions.
- **Parametrization**: refers to the amount and relevance of parameters used in systems. Parameters can be seen as behaviors that are learned or calibrated, controlling the behavior of a technique. Parameters that are calibrated can be used to smooth connections among character occurrences in concatenation systems. On the other hand, parameters can be learned from the data available, sometimes restricting the minimum data required to train the model.

The techniques are further divided into subtopics. Generation techniques can be *perturbation-based*, which consists in altering the input sample in order to obtain new samples, *fusion-based*, which takes two-to-few inputs and fuse parts into novel samples, and *model-based*, which captures variations in writing from many samples into models. Concatenation techniques are classified into *no-connection*, which juxtaposes writing units into text lines, *direct-connection*, which takes writing units and position them and *modeled-connection*, which adds new ligatures by parametric curves.

The work of Haines and colleagues (HAINES; AODHA; BROSTOW, 2016) has detailed the key parts of a handwritten text. Two of the most important terms to known are *glyph* and *ligatures*. A *glyph* is a technical term often used in typography which corresponds to a graphic symbol that represents a writable meaningful character. In other words, it defines a specific instance of a character. The ligature, on the other hand, can be defined in our context as a connector stroke between two glyphs.

As important as the approach itself are the evaluation methods. Evaluations are used not only to assess the quality of a given synthesis approach, but also to assess how well it fulfills the application's final objective. They can be subdivided into objective and subjective evaluations.

Objective methods rely on quantitative metrics on synthesized results. The objective methods can be divided into three types: Optic Character Recognition (OCR) and word retrieval rates, writer identification rates and resemblance with a reference model.

The first two types are evaluated by providing the synthesized results into recognition systems, which are expected to improve the recognition rate because of the newly introduced variability. The last type requires comparing synthesized handwriting to a reference model through correlation and regression analysis.

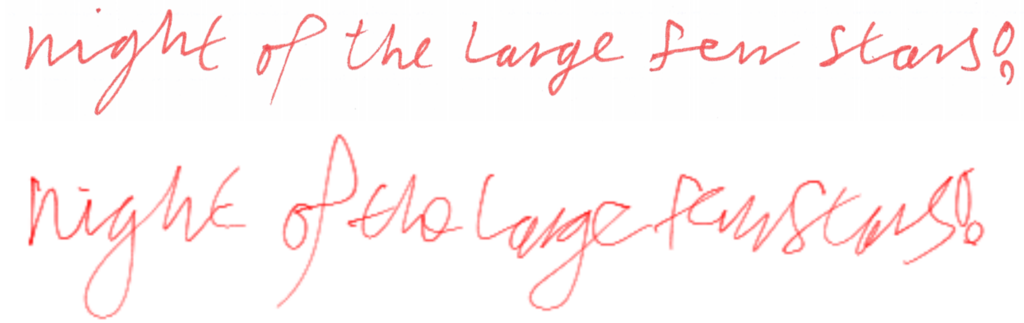
On the other hand, subjective methods rely on the opinion of subjects. Their evaluation types can be classified by non-experts or by handwriting style experts. Although losing the precision compared to objective methods, subjective approaches are suitable to find exaggerated regularities in glyph shapes and eventually some inconsistencies in inking (GUYON, 1996). In order to get the best of both methods, a work may include a combination of subjective and objective evaluations. According to the data extracted from Elarian and colleagues (ELARIAN et al., 2014), subjective evaluation methods appear more frequently in the literature. Unfortunately, there is no standard benchmark for objective comparison among different techniques.

Regarding input data sampling, a common technique is collecting pangrams, which are sentences that use every letter of the alphabet at least once. An example of a famous pangram in English is "The five boxing wizards jump quickly", containing 31 unique letters. In some works, only one sample of each letter is enough as the user's input. This type of technique allows the system to not only capture the letters itself but also information about the ligatures between the characters.

As stated by Haines and colleagues (HAINES; AODHA; BROSTOW, 2016), most online approaches use graphics tablets for input. Even though tablets and similar devices are quite common nowadays, it is a known issue that the writing results are different compared to the traditional ones through pen and paper. This issue can be addressed due to the low friction provided by tablets, producing a distortion. Further, the device usually requires a pen-like specific instrument or even the finger to capture the writing information, which may cause discomfort to the user. Figure 2.1 illustrates the same sentence written by an author using real pen and paper and on a tablet device, showing that, clearly, the one written on the tablet is less readable.

Trying to address this problem, a few companies have developed devices, such as Wacom (WACOM, 2018) and Iskn (ISKN, 2018), that capture the writing of a subject with pen and paper directly, instead of requiring a special tool and writing on a tablet screen. One of these, The Slate, was created by the Iskn corporation (ISKN, 2018), presented in Figure 2.2. One of our main assumptions is that this device provides "the best of both worlds" for written text sampling since it allows for not only online sampling, but also the

Figure 2.1 Two lines written by the same author. The top one uses a real pen and the bottom one a graphic tablet.



Source: Haines et al. (HAINES; AODHA; BROSTOW, 2016).

naturalness of writing on a piece of paper with pen or pencil.

The device consists of a sensitive surface used to recognize and track objects. The pad is equipped with 32 sensors used to detect the position and orientation of a usable object through a magnetic ring. An usable object is defined as an object that contains an Iskn certified magnet, allowing many objects to be used to interact with The Slate. In order to obtain real visual feedback, a paper should be coupled to the main device and fixed through some plastic clips. Also, any common pencil or similar can be equipped with the magnetic ring, allowing the user to write with its own preferable tool. This setup assures that writing or drawing on the device is as normal as daily tasks to the user.

It's important to notice that, although the maximum sampling rate of The Slate is 140Hz, currently we cannot control the sampling rate. There is a moving filter integrated into the firmware of The Slate which depends on the writing velocity. For instance, if the user moves the pen tool slowly, The Slate computes fewer points per second.

The device also includes some main components. There is one RGB led that provides a visual feedback to the user. It also has the power button used to turn it on and off and two buttons which have specific functions when using the official application but may be controlled and assigned to some purposes through the API. The connection can also be made through USB or Bluetooth. Users that are not interested in development may use its official program called Imagink for both mobile and desktop, working like a customized graphics application fully compatible with the hardware.

There is also an Application Programming Interface (API) for developers that use C++ programming language on several platforms. Each sample contains the 3D pen tip position $p = (x, y, z)$. The rotation angles r_x and r_y of the pen tip in hundredth degrees and the acceleration value a on each axis. The angle r_x stands for the angle between the magnet axis and the z axis (from 0^o to 180^o) while the angle r_y stands for the angle

Figure 2.2 Image of The Slate 2.

Source: Iskn website (ISKN, 2018).

between the projection of the magnet axis on the xy plane, and the x axis (from -180^o to 180^o). These hardware details are available in Figure 2.3. The device writable area, called active area, is given in float numbers through the API. Further, it can get values for the z coordinate to show how close the tool is to the surface. Unfortunately, this device provides no way to estimate the pen pressure provided by common tablets.

One of the most common ways to digitally handle text data are computer fonts, which are implemented as digital data files containing glyphs, characters or symbols. They can be divided into three basic categories. Bitmap fonts are matrices of pixels representing the image of each glyph in different sizes. Vector fonts consist of drawing instructions and mathematical formulae to describe the glyphs, making them scalable to any size. Finally, stoke fonts use specified lines and further information to define profile, size and shape of the lines describing the appearance of the glyph. There are two common ways of representing fonts through files. One of them is called True Type Font (TTF) file

Figure 2.3 Image of some hardware properties of The Slate 2. 3D coordinate system (left) and rotation angles (right)

Source: the author.

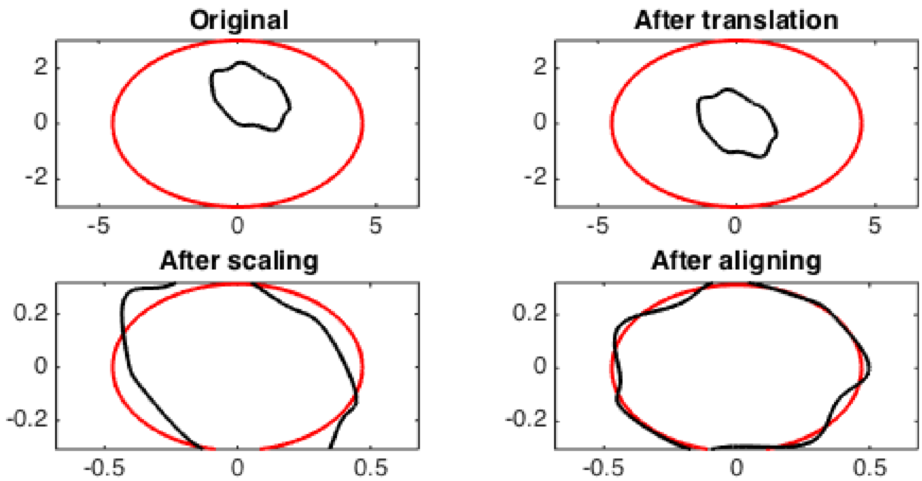
format, while the other one is called Open Type Font (OTF). Both can be classified as vector fonts and have their glyphs described with Bézier curves.

Free fonts are divided into several categories, including the handwritten ones, which can be largely available throughout the Internet. We found that the website Fontspace alone (FONTSPACE, 2018) stores around 3307 results classified as handwriting fonts. Other websites, such as Google fonts (https://fonts.google.com/), Free Calligraphy Wedding (34. . . , 2017) and 1001 fonts (1001. . . , 2017) contain 135, 34 and 1953 handwriting font families, respectively.

Lastly, Procrustes is a form of statistical shape analysis that can be used to analyze the distribution of shapes. This technique optimally translates, rotates and uniformly scale objects in order to obtain a similar placement and size by minimizing a measure of shape difference between them. The objects are described as a set of points and usually the comparison occurs between two shapes. Once the shapes are superimposed, the algorithm returns its distance as a real number between 0 and 1. The closer to zero, the more similar the shapes are. Figure 2.4 presents and example of the Procrustes usage.

In this chapter, we covered some main concepts of the handwriting synthesis field, besides some details regarding aspects which will be used in the next chapters. Our exposition included some relevant points towards evaluation methods, data sampling and handwriting fonts.

Figure 2.4 Example of the Procrustes algorithm process for two different shapes.



Source: https://www.chebfun.org/examples/geom/Procrustes.html.

3 RELATED WORK

In this chapter, we present the most relevant research on the handwriting topic. We starting by covering the handwriting synthesis most relevant works in the first three sections. We followed the data sampling classifications proposed by Elarian and colleagues (ELARIAN et al., 2014), dividing them according to its data sampling type. Next, in Section 3.4, we investigate some of the shape matching techniques which could be fitted in the handwriting area. Finally, our last section reviews the handwriting databases publicly available. Figure 3.1 presents an overview of the related works.

Figure 3.1 Summary of the related works reviewed in this thesis.

Online		Offline		Mixed	Shape Matching	Handwriting databases
Zheng et al. (2005)	Guyon (1996)	Helmers et al. (2003)	Font websites (2019)	Miyao et al. (2005)	Veltkamp (2001)	UNIPEN (1994)
Stettiner et al. (1994)	Lin et al. (2007)	Thomas et al. (2009)	Nakamura (2018)	Liu et al. (2011)	Chambers et al. (2018)	IAM-OnDB (2005)
Choi et al. (2004)	Rao (1993)	Elarian et al. (2015)	Haines et al. (2016)		Bai et al. (2008)	IRONOFF (1999)
Jawahar et al. (2006)	Wang et al. (2002, 2005)	Fujioka et al. (2006)			Torsello et al. (2004)	CEDAR (1994)
Saabni et al. (2009, 2013)	Graves (2013)					CENPARMI (1996)
						IAM-Database (2002)

Source: the author.

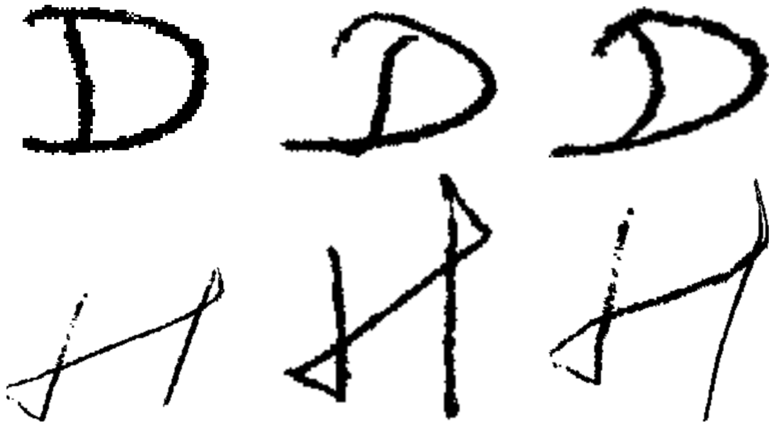
3.1 Online Data Handwriting Synthesis Techniques

We discuss in this section the most relevant online approaches, i.e. techniques that capture time information when collecting input data.

Zheng and colleagues (ZHENG; DOERMANN, 2005) present a point-matching algorithm aiming to learn the shape deformation characteristics of handwriting samples. The handwriting sample is composed of a set of points uniformly sampled from its skeleton. After the matching, they use the Thin Plate Splines (TPS) to control the deformation between the handwriting samples. They claim that TPS can be used for representing coordinate transformations because it is parameter-free with a physical explanation and closed-form representations. The work creates online Latin letters by arranging the points between two samples. Figure 3.2 presents some results achieved with their technique.

Stettinger et al. (STETTINER; CHAZAN, 1994) work with the script's vertical and horizontal velocities, which are modeled as the impulse response of a 2D linear time-varying second-order system. The parameters of the model are estimated by using an analysis-by-synthesis approach, with its distortion measured as a Weighted Least Squares

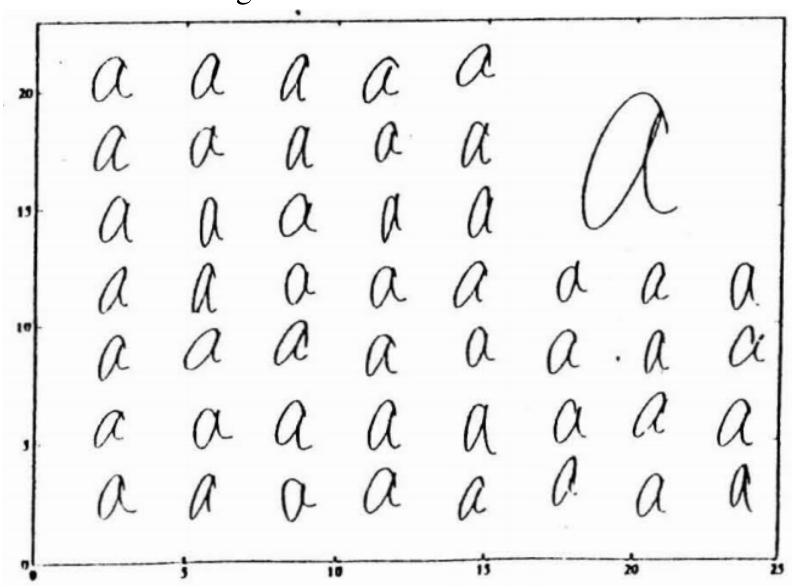
Figure 3.2 Real and synthesized handwriting samples from Zheng et al. Left and middle columns: real samples. Right column: synthesized samples.



Source: Zheng et al. (ZHENG; DOERMANN, 2005).

of the difference between the original and synthesized script. Then, the model's parameters are estimated by solving a nonlinear optimization problem using the Modified Newton Method. They also present a statistical model using a parametric representation for all the letters of the alphabet. This method produces a large number of characters, enhancing its variability. Figure 3.3 shows a few characters synthesized using the technique.

Figure 3.3 Random synthetic characters, created by synthesizing a parameter vector, randomly chosen from a cluster, represented by its mean vector (enlarged) and covariance matrix from the work of Stettinger et al.

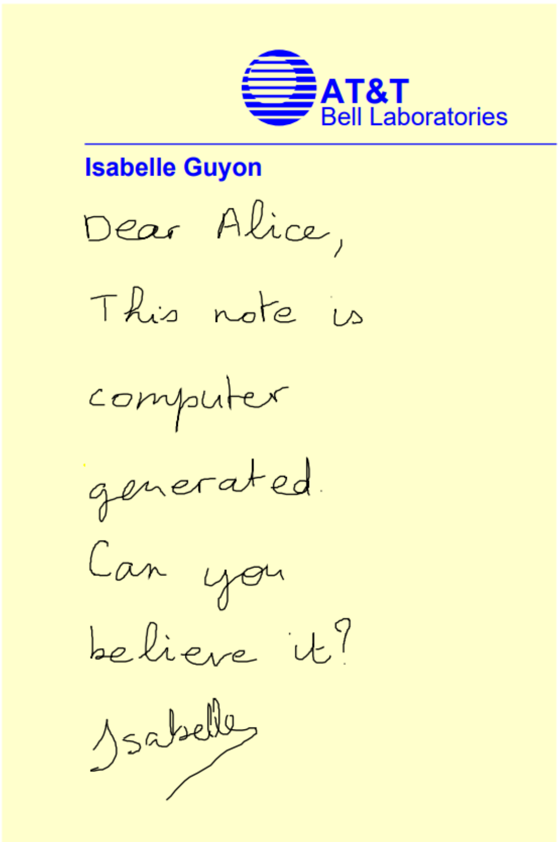


Source: Stettinger et al. (STETTINER; CHAZAN, 1994).

Choi and colleagues (CHOI; CHOI; KIM, 2004) generate character shapes from handwriting recognizers based on Bayesian networks. First, they use different handwriting samples to train handwriting online recognizers. Next, they generate character shapes from texts to find the most probable sequence of input points. The authors claim that Bayesian network based classifiers have many parameters for modeling components since they produce more natural shapes than working with Hidden Markov Models (HMMs). Experiments comparing their approach with HMMs for Hangul and digits generated more natural character shapes.

Guyon (GUYON, 1996) introduced in 1996 a simple technique to render handwritten texts assembling glyphs from samples collected from the end user. The main idea is to juxtapose selected string images containing a sequence of letters previously captured from the user. An application which allows the user to enter typed text and obtain handwritten note with his own handwriting is provided to the user. The synthesis results are accurate, but the technique needs as input a long list of 100 3-letter entries, called *lexicon* to be handwritten by the user. Also, the lexicon was targeted for English, and it is not clear how it would work with other languages. Figure 3.4 presents a result of this technique.

Figure 3.4 Example of handwriting synthesis from Guyon with a thin brush and black ink.



Source: Guyon (GUYON, 1996).

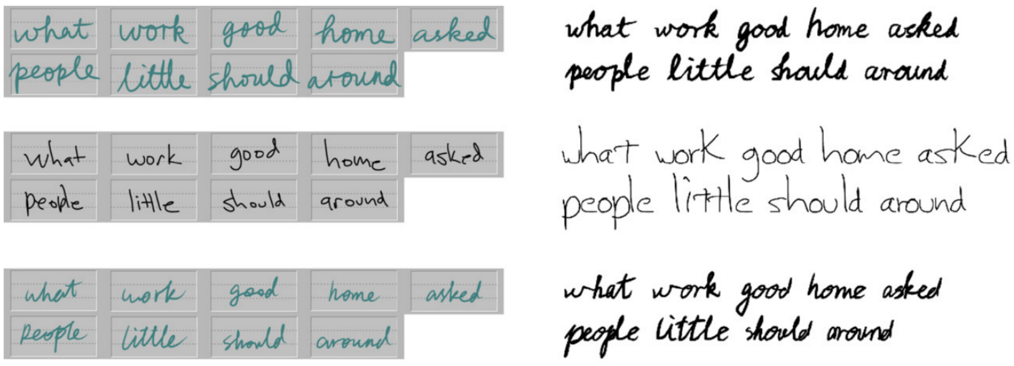
Jawahar and Balasubramanian (JAWAHAR; BALASUBRAMANIAN, 2006) detailed a synthesis model for generating handwritten data for Indian languages. The authors argue that the spatial layout of components are simple with respect to the English language. The handwriting model consisted of two parts: a stroke model, capturing the shapes and variations of characters for each stroke, and a layout model, controlling the spatial layout of individual strokes. Unfortunately, the paper is focused only on Indian languages.

Saabni and El-Sana (SAABNI; EL-SANA, 2009; SAABNI; EL-SANA, 2013) presented a technique to generate databases by synthesizing prototypes for each word in a lexicon including different appearances of the same letters. Their work has several resemblances with Guyon's (GUYON, 1996), however, they target Arabic. Dimensionality reduction and clustering were applied to represent the databases compactly. The authors approach is divided into three main procedures. Firstly, sets of different handwriting prototypes are produced by allowing users to enter words that cover a large number of glyph position and shape variability. Then, they generate multiple shapes for each word-part in a lexicon by concatenating the letters, producing a large amount of possible shapes for each word-part. Finally, the authors divide the shapes of each word-part into groups using clusters, selecting a small set of representative shapes for each group, decreasing the database size. Other scripts, such as Latin, can be done following different rules for connecting letters, although this does not seem straightforward. They also aim at generating an offline dataset as output, which differs from our approach.

Lin and Wan collected features about a particular handwriting and used these to synthesize new text in a hierarchical fashion (LIN; WAN, 2007). Variation is introduced at the character level, and the sampling of input text uses a specially designed interface. In a first moment, they sample characters and add shape variation. After, they generate words by aligning the glyphs with proper horizontal and vertical spacing. In order to avoid overlap and to facilitate the connection between letters, the beginning and ends of the glyphs were trimmed and adjacent letters were connected by polynomial interpolation. They also presented a user-input study consisting of eight testers. Some users did not have experience in tablet writing, taking around 20 minutes to complete. An integration with Microsoft Office Outlook was also implemented, allowing the user to type the desired text on the keyboard and convert it to an image representing the handwriting. Although they produced high-quality output, the system assumes that the users write at constant speed, making the handwriting to eventually become difficult to read. Also, their input method requires the user to fill a specific form, which is less natural than a sentence, for instance. Figure 3.5 shows some results of their synthesis technique.

The work by Rao (RAO, 1993) synthesized characters as a collection of straight

Figure 3.5 Results from the work of Lin and Wan. Handwriting samples (left column) from some writers and their respective synthesized handwriting (right column).



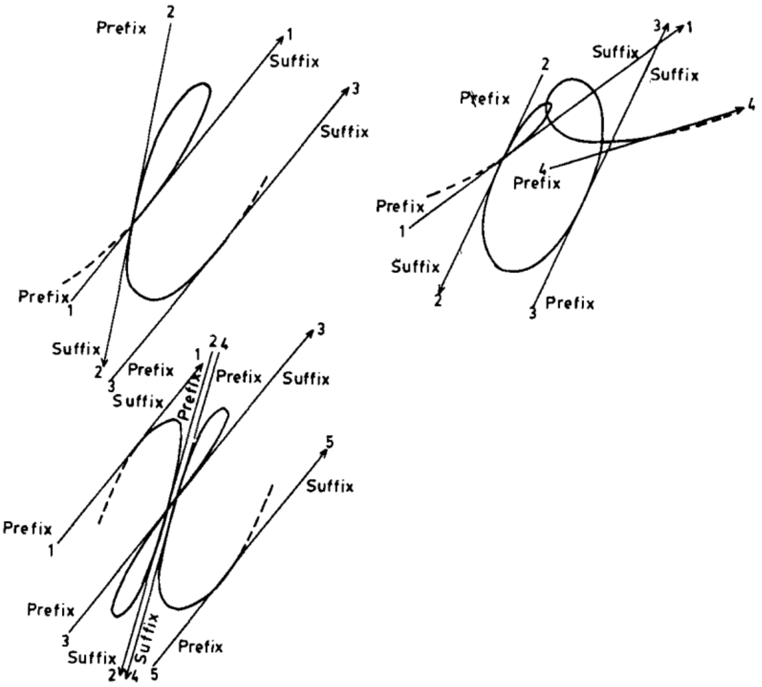
Source: Lin and Wan (LIN; WAN, 2007).

line segments called shape vectors approximated from a sample of a real written character. This is possible since characters may have shapes which are straight segments alternating with regions of high curvature. The concatenation part connects the end of a letter to the beginning of the next one through Bézier curves. The end segments of the characters are defined as the control points, making the resulting Bézier curve to represent the transition segment. The author claims this transition segment replicates the characteristics of a transition in natural connecting writing. Assuming the segments of letters were correctly captured, the results appear natural. The author provided the fit distance between the original and the synthesized characters as an evaluation metric. Further, it provided a data reduction ratio by representing these characters through vectors. Although it presented good results overall, the work remained restricted to characters. The technique also depends on parameters for the setting of character shapes. Further, the work can be seen as a metric to encode recognition inputs to curves, without introducing user variability. Figure 3.6 present some characters synthesized with shape vectors.

Wang and colleagues (WANG et al., 2002) presented a two-step learning model on the points describing a B-spline curve for each sampled character. They explored the idea of tri-units considering that each character, in general, is surrounded left and right by neighbors that affect how the cursive writing flows. Their two-step learning model consists of a template-based matching algorithm and a data algorithm in order to extract training vectors from the samples. Then, in the synthesis process, letters and ligatures are generated from learned models and further concatenated to produce the word trajectory, guided by a deformable model.

A learning approach combining shape models and physical models was presented by Wang and colleagues (WANG et al., 2005). They used as input text written on a

Figure 3.6 An example of the letters ‘e’, ‘o’ and ‘x’ synthesized through shape vectors in the work of Rao.



Source: Rao (RAO, 1993).

tablet. Sample paragraphs are collected and further segmented into individual characters by using a two-level writer-independent segmentation algorithm. The samples of the same character are classified into a coordinate frame and learned by shape models, in order to generate shapes similar to the training set. Then, for the synthesis process, individual letters are generated from models and concatenated by a conditional sampling algorithm in order to generate natural cursive handwriting. As with many of the other approaches here revised, the validation is only subjective, by visual comparison between input and output. Their approach is limited by the number of samples used for training because the shape models only produce new shapes within the variation of training samples.

The approach from Graves (Graves, 2013) uses a long short-term memory recurrent neural network to produce sequences with long-range structures by predicting one data point at a time. They extend their technique to the field of handwriting synthesis by allowing the network to condition its predictions on a text sequence. They used the data from the IAM online handwriting database (LIWICKI; BUNKE, 2005), which is further described in the Section 3.5. Their technique, in particular, requires minimum preprocessing and feature-extraction methods, since they wanted the network to model the data as raw as possible, instead of reducing the variation in the data.

Since their approach predicts the pen traces one point at a time, it gives their network maximum flexibility to create novel handwriting. After the algorithm successfully predicts the handwriting, they extended their technique to handwriting synthesis, allowing the network to learn which character to write next. They produced results that are convincing and look natural. However, they present a very sophisticated approach with neural networks besides using a vast database to train their neural network, containing 5364, 1438, 1518, and 3859 lines in the training set, two validation sets, and a test set respectively.

Recently, the work of Askan and colleagues (AKSAN; PECE; HILLIGES, 2018) proposed a generative neural network architecture for disentangling style from content, making digital ink digitable. Their model allows an user to control over the visual style, similar to beautification techniques. Similarly to the previous reviewed work, they also require a large amount of data. Since they collect online handwriting using a tablet, they also do not provide haptic feedback given by real pen and paper.

3.2 Offline Data Handwriting Synthesis Techniques

In this section, since our offline solution is classified as a concatenation approach we, therefore, review here only these approaches. We also give emphasis to the current state-of-the-art research on offline data handwriting synthesis.

Helmers and Bunke defined three different methods for synthesis of handwritten text (HELMERS; BUNKE, 2003) to be used in the context of handwriting recognition. They show that the recognizer usually performs as well as or slightly better than using only real handwriting. In the context of CAPTCHA generation, Thomas, Rusu, and Govindarauj (THOMAS; RUSU; GOVINDARAJU, 2009) introduced artificial CAPTCHAS that look like they were handwritten but without being writer-specific. Their generated CAPTCHAS performed better against automated bots. Some methods target specifically other script languages (ELARIAN et al., 2015), or Japanese (FUJIOKA et al., 2006) glyphs.

There are many websites (CALLIGRAPHR, 2017; MYSCRIPTFONT, 2017; YOURFONTS, 2017) that generate a font family from a user's handwriting. The user writes the alphabet of characters, one by one, on a given template grid which is later digitized and used as input for computing the family font. However attractive, these approaches suffer from the fact that the writing of individual characters makes the process artificial and the results often look unnatural, being mostly aimed for artistic purposes. Some of these results can be seen in Figure 3.7, showing several excerpts extracted from different fonts of the Calligraphr website. There is no doubt the information is presented as a pure artistic approach for creating its own personalized font, which does not intend to represent the user's calligraphy. Further, the input template approach represents a poor solution to capture an user's handwriting. Closely related to our work is the website WhatTheFont (WHATTHEFONT, 2017). From an image of a particular character, they provide 10 suggestions of similar looking characters. However interesting, many suggestions do not provide good matches. Besides, it is a black box without any information on how exactly the matches are computed, and it seems to only suggest families from their own web service.

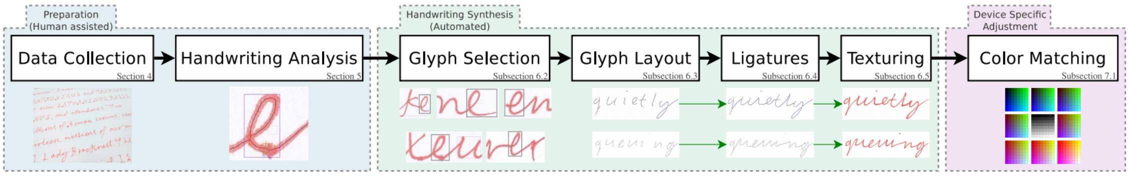
Figure 3.7 Examples of fonts constructed by the users on the Calligraphr website.

Source: Calligraphr website. (CALLIGRAPHR, 2017).

Nakamura and colleagues (NAKAMURA et al., 2018) presented a more recent offline approach. They introduced HCC - Handwritten Character Clones, that is, visually similar characters from a given person's handwriting. They used as input images of Japanese characters, and their contribution focused on how to synthesize images of characters even for incomplete sets, that is, sets where at most one or no image is available for some characters.

The current state of the art is an offline shape simulation algorithm recently proposed by Haines and colleagues (HAINES; AODHA; BROSTOW, 2016). From a sample of a user's handwriting, the objective is to render a realistic handwriting in the same style of the writer's input. The main idea is to reuse glyphs provided by an author and processed through an interface in order to provide believable output. According to their pipeline, presented in Figure 3.8, their algorithm is divided into three main phases, called preparation, handwriting synthesis and post-processing.

Figure 3.8 Pipeline of the approach used by Haines et al.



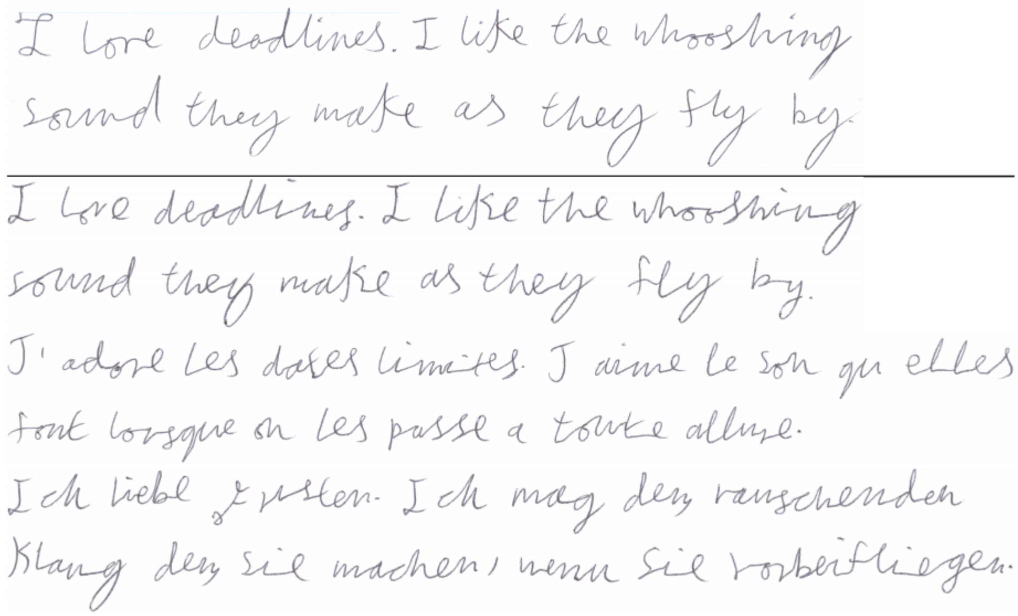
Source: Haines et al. (HAINES; AODHA; BROSTOW, 2016).

The first phase is human-assisted and consists of two steps. The first step is called data collection and consists of capturing the user's handwriting. When the author is available they request the author to write four A4-sized pages with some pre-defined text. The idea here is to collect a large combination of distinct characters and ligatures to increase the variability for the synthesis. If the author is not available, however, they can only synthesize the characters available in the input sample. The following step provides the handwriting analysis, which is divided into automatic tagging and human assistance to correct the previous task. In the automatic tagging, the goal is to segment the ink and tag pixels according to the source text letters. The tagging approach assigns a label, indicating the letter the glyph represents, a split, indicating the point where the line transitions from glyph to ligature and a link, indicating that two separate lines are part of the same glyph. They point out that compared to normal handwriting this is an unusual scenario, since the source text is already known, but the transition points are unknown.

The second phase is automated, and it is divided into four steps. The problem is treated as an optimization problem and the objective is to minimize the cost function. First, they present their glyph selection algorithm. Given a database with multiple scanned samples of an author's handwriting, they choose the glyphs according to some factors, such as the original position in the sentence and the number of times it was already used. Next, they deal with the spacing of the glyphs, which consist of optimizing the position of the glyphs on the page in order to match the author's style in terms of vertical and horizontal spacing between adjacent glyphs. The third step treats the ligature problem, selecting optimally the best-fitting ligatures between glyphs according to the author's style. The last step is called texture rendering, and it minimizes the difference between the original glyph texture and the output image, trying to ensure continuity of the pen path.

The last phase is called post-processing and the idea here is to render realistic handwriting not only on screen but also on paper. They consider the color matching of the output printer in order to convince the users that the results are in fact written directly with pen and paper instead of being printed. Figure 3.9 shows one of the results of the approach.

Figure 3.9 Example of the handwriting synthesis algorithm from Haines et al. The real writing is on top and the three below are computer generated, preserving the same style.



Source: Haines et al. (HAINES; AODHA; BROSTOW, 2016).

They obtained a dataset containing several handwriting samples from nine different subjects using different pens and inks. They chose a subset of the dataset to validate the system representing the whole. The authors created three study hypotheses to validate their approach: (i) the synthesized handwriting results are realistic, (ii) the synthesized handwriting is in the same style as an author and (iii) the synthesized handwriting when printed, is realistic if compared to a real pen. In order to evaluate the first two hypotheses, 170 participants were assigned with a printed questionnaire. For the realism evaluation, the participants were assigned to 10 sentences and asked for each of them if it was computer generated or not. For the style synthesis evaluation, the participants were given a real sentence and two other samples with the same characteristics, which one was real and the other one generated by the algorithm. Finally, for the calibration evaluation, the participants were asked to determine for a sequence of printed envelopes if they were written by a human using a pen or if it was printed by a computer.

Although they present natural looking results, they claim offline approach was preferred over online since online techniques need to handle the tablet acquisition distorting problems. They also require a large number of words as a text to produce good results because their result depends on the variability learned from each user on input.

3.3 Online and Offline Data Type Techniques

Some approaches can deal with both online and offline data types. In this section, we discuss some of them.

Miyao et al. (MIYAO et al., 2005) proposed a method for improving the offline character classifiers through accessing the virtual examples already synthesized from an online database. Since they claim a large database is required to obtain good classifiers, they propose a method to train Support Vector Machines (SVMs) for offline character recognition based on artificially augmented examples using online characters. Simple affine transformations are applied to the strokes in order to generate virtual samples. The results showed that strokes are better than characters regarding the transformation when generating training samples for SVMs.

Hidden Markov Models (HMMs) were used as a generative statistical model to synthesize handwritten samples by Liu and colleagues (LIU et al., 2011), which patented the idea. Then, the trained HMMs can be adapted through some technique such as a maximum linear regression technique.

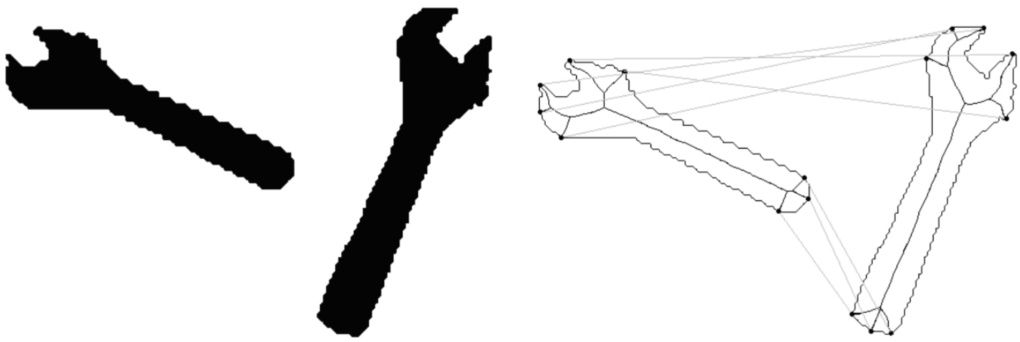
3.4 Shape Matching

Since handwriting similarity can be considered a particular case of the shape matching problem in general, we also investigated some of the most common techniques related to 2D shape comparison. An overview of the topic is presented by Veltkamp (VELTKAMP, 2001), which covers several aspects related to solving shape matching problems, like selecting the properties of the similarity measure to solve the problem, and constructing the algorithm to compute the measure effectively. A large part of the techniques tends to focus on skeleton detection and comparison.

The recent work of Chambers *et al.* (CHAMBERS et al., 2018) explores different notions of shape complexity by applying them to a library of shapes using k-medoids clustering. They divide the measures into three main categories: skeleton-based, symmetry-based, and boundary sampling. The idea is to discover what aspects of shape complexity are captured by each notion. Their results show that each of the measures provides important information about a shape's complexity.

Bai and Latecki (BAI; LATECKI, 2008) propose a graph matching algorithm applied to shape recognition based on object silhouettes. They compare the geodesic paths between skeleton endpoints to match skeleton graphs. They believe that visually similar skeleton graphs may be completely different regarding their topological structures, using an approach which partitions the skeletons with Discrete Curve Evolution. Since they do not use any tree/graph matching, they reduce the time cost of similar methods. An example of their graph matching result can be seen in Figure 3.10.

Figure 3.10 Example of the correspondence skeletonization graph computed from two distinct shapes of a wrench.



Source: Bai and Latecki (BAI; LATECKI, 2008).

The work of Torsello and Hancock (TORSELLO; HANCOCK, 2004) presents a geometric measure to gauge the similarity of 2D shapes by comparing their skeletons. Their measure is defined to be the rate of change of boundary length with distance along the skeleton. This approach varies continuously even when the shape deforms.

These techniques aim to match the geometry and topology of the compared shapes globally, being time-consuming (LEONARD et al., 2016). That is expected since they need to achieve good results for a large number of shape possibilities. For a more complete review of the vast area of skeletonization and graph matching algorithms, please refer to the recent surveys on the field (SAHA; BORGEFORS; BAJA, 2016) (FOGGIA; PERCANNELLA; VENTO, 2014).

3.5 Handwriting Databases

In this section we describe some of the handwriting databases available in the literature. Most of the databases are focused on providing data for handwriting recognizers, such as neural networks, support vector machines or Hidden Markov Models (HMMs), since they all require information to be trained. However, these databases also contribute for the handwriting synthesis field by providing input samples to produce the new ones. Databases can usually be grouped in online, offline or even both and are mostly available for the English language.

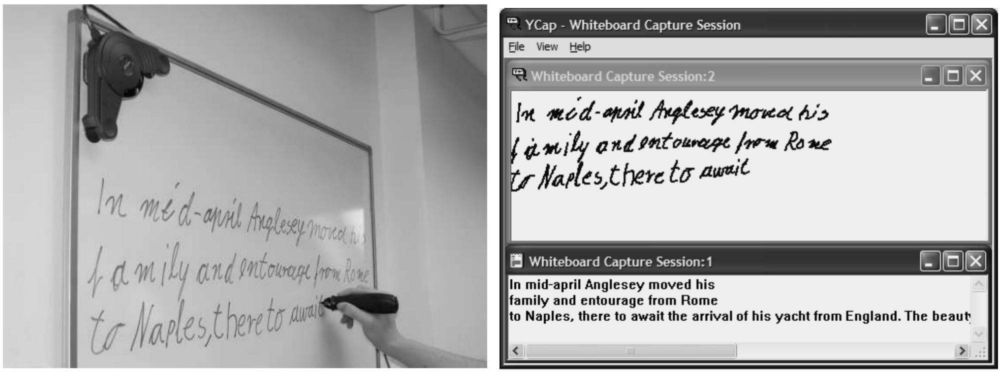
Online databases include the UNIPEN (GUYON et al., 1994), which is a large online handwriting database containing isolated characters, single words and some sentences. The purpose of their project is to propose and implement solutions to the growing demand for handwriting samples. Also, the minimum number of required signal channels is x and y, but they encouraged more signals, such as pen angle or pressure information.

A very interesting online database called IAM-OnDB (LIWICKI; BUNKE, 2005) is defined as an online English sentence database acquired from handwritten text on a whiteboard. The authors provide an approach for meeting rooms, where whiteboards are largely used to discuss and register ideas. These areas typically have multiple microphones and video cameras to record a meeting. It consists of handwritten lines collected from 221 different writers using a smart whiteboard, containing around 86 thousand word instances. All texts included in their approach are taken from the Lancaster-Oslo/Berger corpus (LOB), which is an electronic corpus of text (JOHANSSON et al., 1986). The position of the pen was tracked with an infrared device at the corner of the board, capturing the x and y pen coordinates along with a time stamp for each location and the points in sequence when the pen is lifted off the whiteboard. Their output data contains the transcription produced by the operator together with the online data recorded in a single xml file. Figure 3.11 shows a recording illustration and its respective capture software. They are the first work dealing with capture on whiteboard, allowing for new contributions using new hardwares instead of tablet-only approaches.

IRONOFF (VIARD-GAUDIN et al., 1999) is an online and offline database which contains the scanned images handwritten words in both English and French. This is an interesting feature since the authors can merge the online information along with the offline texture, for instance.

Some offline databases include CEDAR (HULL, 1994), for postal address recog-

Figure 3.11 Recording illustration (left) and interface of the recording software (right).



Source: IAM-OnDB (LIWICKI; BUNKE, 2005).

nition, CENPARMI (LEE, 1996), a database exclusive for handwritten numerals, and the IAM-Database (MARTI; BUNKE, 2002), which presents a large collection of unconstrained handwritten sentences.

Other scripts besides Latin also have their own databases through research papers, such as Chinese (JIN et al., 2011), Arabic (ABED et al., 2009) and even multiscript (NJAH et al., 2012) (DJEDDI et al., 2014).

3.6 Discussion

We presented in this chapter related techniques regarding handwriting synthesis, followed by some of the most relevant works in the shape matching field. We also discuss some of the handwriting databases available on the web. The reviewed works stated that handwriting synthesis problems are still open, allowing them to be further explored. We showed that the reviewed online databases store data usually from tablets, which is the most prominent data acquisition device. This device typically provides no friction, producing an artificial distortion. We also believe public fonts have potential to be explored for handwriting synthesis, since a very large amount is publicly available on the web. We observed that the state-of-the-art approaches for handwriting synthesis deal with a large amount of data and are time consuming, giving space for further contribution on synthesizing good results requiring less input data and processing.

4 ASSESSING SIMILARITY IN HANDWRITTEN TEXTS

Current state-of-the-art handwriting synthesis requires the subject to spend around 40 minutes writing the samples (HAINES; AODHA; BROSTOW, 2016). The primary motivation for such long capture time is the need to capture variations in the real handwriting such that the synthesis can later use this variation. However, the current amount of input samples is defined in a conservative approach, without answering the question of how many samples are needed to reproduce natural variability. An alternative to collecting long texts is to collect pangrams. The argument against pangrams is that they do not capture enough variation about a subject's handwriting. Therefore, some techniques approach this problem by asking the user to write a reasonable amount of text to collect variability, as in (HAINES; AODHA; BROSTOW, 2016). However, the question regarding the variability of pangrams against longer texts still remains unanswered.

In this chapter, we report on a study we have conducted comparing written text similarity between two sets of samples, one using augmented pangrams (with a total of 473 characters) and the other using general texts (with 1586 characters). The discoveries regarding this research allow future works to decrease or maintain the number of samples to be collected. Further, we also used the tablet-like hardware called The Slate (ISKN, 2018), which was already presented in Chapter 2, to capture handwriting samples.

The first section describes our methodology, which is subdivided in four steps. Next, in the second and last section we present our results.

4.1 Methodology

We conducted a user study to answer how the amount of text written by a subject is directly related to its variability. This section is divided into four parts. The first part introduces our data acquisition process, followed by the segmentation, shape matching, and data validation steps of our methodology.

4.1.1 Data Acquisition

Our data acquisition step uses the hardware called The Slate (ISKN, 2018), previously described. We asked the subjects to write the samples using a pencil in A5 sheets of blank paper attached to the hardware. Underneath each sheet, we placed a blank guide sheet with bold lines. To minimize text mismatch, we advised the subjects to follow these lines. Each subject copied the texts from two sets of samples. We alternate the sample writing order between sets for each subject to avoid bias.

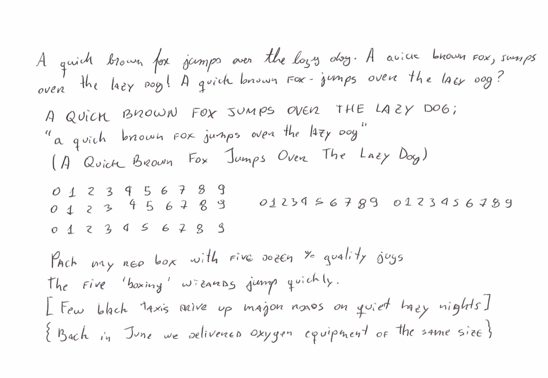
The first set, which we call *Universal*, consists of an entire text plus text fragments extracted from the work of Haines et al. (HAINES; AODHA; BROSTOW, 2016). They used an algorithm to generate sample texts using fragments from English literature. Among the generated texts from their work, we chose the one with the largest letter variability and the least amount of characters in general. Since no text had at least one sample of each letter, we extracted a few parts from other texts that contained the remaining letters and appended them to the chosen text, all of them extracted from Haines et. al. This first set has a total of 1586 characters (not counting blank spaces). The second set, which we call *Pangram*, consists of a small set of English pangrams and digits (numbers from 0 to 9), where we introduced some symbols (dot, comma, colon, etc.) and added the upper and lower case of letters. This sample has a total of 473 characters (also not counting blank spaces). Considering the number of characters, the Pangram set is around 30% the size of the Universal one.

The subject starts the experiment by answering an online pre-questionnaire containing the following information: condition (whether it started by writing the Universal or the Pangram set); age; education level; gender; how often the subject writes by hand (hours per week); dominant hand for writing; if the subject is an English native speaker, and his English reading, writing and speaking English skills. The subject had no time constraints for completing the task and could rest or stop whenever desired.

Next, we give the subject a printed version of both the instructions and the samples following its condition order (Universal or Pangram), allowing the start of the experiment. For each time the pencil touches the hardware, we collect its 2D point coordinates (x_i, y_i) with a respective timestamp t_i. During the writing, the subjects were instructed to cross out a given word if they were not satisfied with it. After the subject finishes writing each page, we store the subject's data, change the paper, and repeat the procedure until the end of the set. During the handwriting acquisition process, the users can follow their writing onscreen in real-time as a rendered image. That could be used as alternative feedback in case the subject wants to check how the hardware was capturing its information. The output of this step is a CSV text file for each subject, where each line contains three floats (x_i, y_i, t_i) separated by commas.

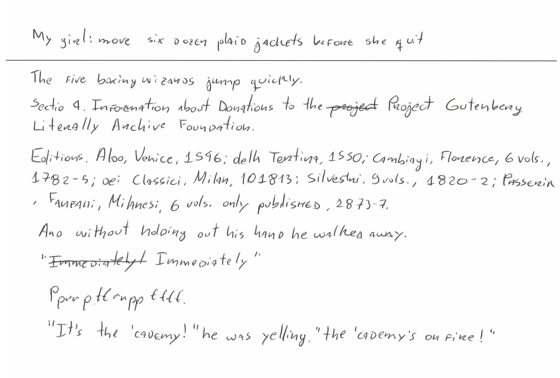
Figure 4.1 Scanned sample texts from subject 07. We introduced a line to divide the Pangram and the Universal texts.

(a) Subject 07 - Page 1



A quick brown fox jumps over the lazy dog. A quick brown fox, jumps over the lazy dog! A quick brown fox - jumps over the lazy dog?
A QUICK BROWN FOX JUMPS OVER THE LAZY DOG;
"a quick brown fox jumps over the lazy dog"
(A Quick Brown Fox Jumps Over The Lazy Dog)
0 1 2 3 4 5 6 7 8 9
0 1 2 3 4 5 6 7 8 9 0123456789 0123456789
0 1 2 3 4 5 6 7 8 9
Pack my red box with five dozen % quality jugs
The five 'boxing' wizards jump quickly.
[Few black taxis drive up major roads on quiet hazy nights]
{ Back in June we delivered oxygen equipment of the same size}

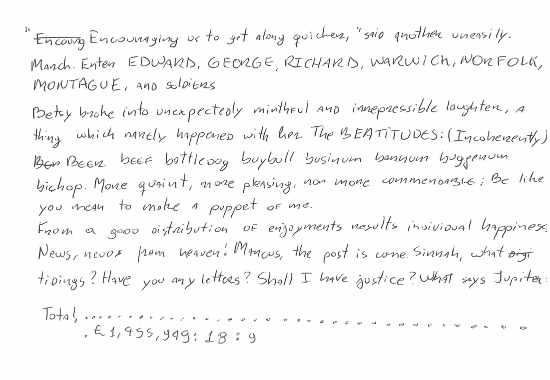
(b) Subject 07 - Page 2



My girl: move six dozen plaid jackets before she quit

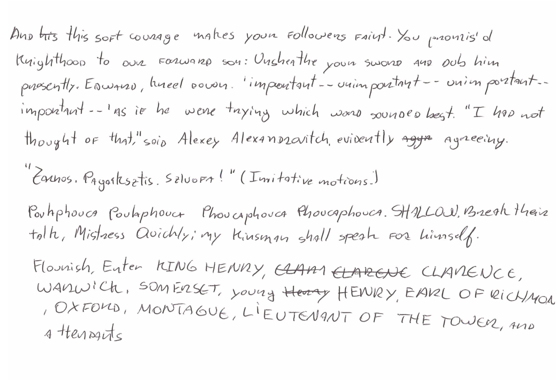
The five boxing wizards jump quickly.
Sectio 4. Information about Donations to the ~~project~~ Project Gutenberg Literally Archive Foundation.
Editions. Aldo, Venice, 1546; della Tentina, 1550; Cambiagi, Florence, 6 vols., 1782-5; dei Classici, Milan, 101813; Silvestri, 9 vols., 1820-2; Passerin, Fanfani, Milanesi, 6 vols. only published, 2873-7.
And without holding out his hand he walked away.
"~~Immediately~~ Immediately"
Pprrpttrrupp tttt.
"It's the 'cademy!" he was yelling, "the 'cademy's on fire!"

(c) Subject 07 - Page 3



"~~Encoura~~ Encouraging us to get along quicker," said another uneasily.
March. Enter EDWARD, GEORGE, RICHARD, WARWICK, NORFOLK, MONTAGUE, and soldiers
Betsy broke into unexpectedly mirthful and irrepressible laughter, a thing which rarely happened with her. The BEATITUDES: (Incoherently) ~~Ber~~ Beer beef battledog buybull businum barnum buggerum bishop. More quaint, more pleasing, nor more commendable; Be like you mean to make a puppet of me.
From a good distribution of enjoyments results individual happiness.
News, news from heaven! Marcus, the post is come. Sirrah, what ~~digi~~ tidings? Have you any letters? Shall I have justice? What says Jupiter:
Total, ...
. £1,455,949: 18 : 9

(d) Subject 07 - Page 4



And ~~his~~ this soft courage makes your followers faint. You promis'd Knighthood to our forward son: Unsheathe your sword and dub him presently. Edward, kneel down. 'important -- unimportant -- unimportant -- important --' as if he were trying which word sounded best. "I had not thought of that," said Alexey Alexandrovitch, evidently ~~agre~~ agreeing.
"Zarnos. Pagarkszetis. Szluoра!" (Imitative motions.)
Poulaphouca Poulaphouca Phoucaphouca Phoucaphouca. SHALLOW. Break their talk, Mistress Quickly; my kinsman shall speak for himself.
Flourish. Enter KING HENRY, ~~CLAR~~ ~~CLARENE~~ CLARENCE, WARWICK, SOMERSET, young ~~Henry~~ HENRY, EARL OF RICHMOND, OXFORD, MONTAGUE, LIEUTENANT OF THE TOWER, and attendants

Source: the author.

After the subjects complete the task, we ask them to answer a quick post-questionnaire containing the following information: how natural was the writing on The Slate compared to common pen and paper; how experienced the subject is on using tablet devices; and how natural was the writing compared to tablet devices (only in cases that the subject has used tablets at least once). For each of the quantitative answers, we offered a scale from 1 to 5, where 1 is the most negative and 5 the most positive. The idea is to gather information about the hardware and its impact on handwriting, minimizing the problem of online writing distortion when using tablet devices. We also tracked the time to complete the task, measuring how long it took each user to complete the experiment by sampling the writing time directly from the device. We analyze this material in the results. Fig. 4.1 illustrates the complete handwriting for subject 07. The user started with the Pangram set, followed by the Universal one.

Together with the 2D points and time stamp data for each subject, we also digitized the written pages to make them publicly available. We believe this is a strong contribution since it represents data available for research for both online and offline synthesis and recognition techniques. We address the questionnaire answers provided by our subjects in Section 4.2. The complete experiment template along with the pre and post questionnaires are available in Appendices A and B.

4.1.2 Segmentation

After our data sampling is complete, we start the segmentation process. We opted for a user-assisted segmentation since there is still no fully automatic algorithm available in the literature, and we need to process information only for a few subjects. We built a segmentation tool to aid the process since we are dealing with thousands of points per subject. Our program reads several comma-separated values (CSV) files with the three float values mentioned before (x_i, y_i, t_i) per line, each one corresponding to a point drawn by the subject in a single page. The goal of this step is to produce a new CSV file in which we assign a unique label for each point. In the end, all sets of points that have the same label belong to the same group, i.e., they correspond to the same glyph. A glyph is a technical term used to define a specific instance of a character.

Our user interface renders all points written by the subject and allows the application's user to inspect each point in the same order it was captured in The Slate. For each point, one can assign a label either by typing it or by pressing specific keys on the keyboard. A shortcut option allows creating a unique automatic label for each upper or lower letter plus any digit by pressing its equivalent key on the keyboard. We also labeled the 17 special symbols, ligatures and an empty label. The list of special symbols along with their labels are available at Table 4.1. When a shortcut label is pressed, our program increments its counter, generating a unique label for a set of points. The empty label is the only exception to this rule since it is used to erase the points, for instance, when a user miswrites a word and strikes it through.

Once the label is decided for a point, the user can replicate the selected label to its subsequent points, navigating quickly through a set of points. This feature shortens significantly the time spent since, for, in most cases, the points are adjacent for each glyph. The point-label association is 1-on-1, i.e., each point is related to one label only.

For each input CSV file, we produce a new one with the original three float values per point plus a $label_i$ as the fourth parameter. It is essential to highlight that this step requires every point to contain a label, including the empty ones. Finally, we divide each

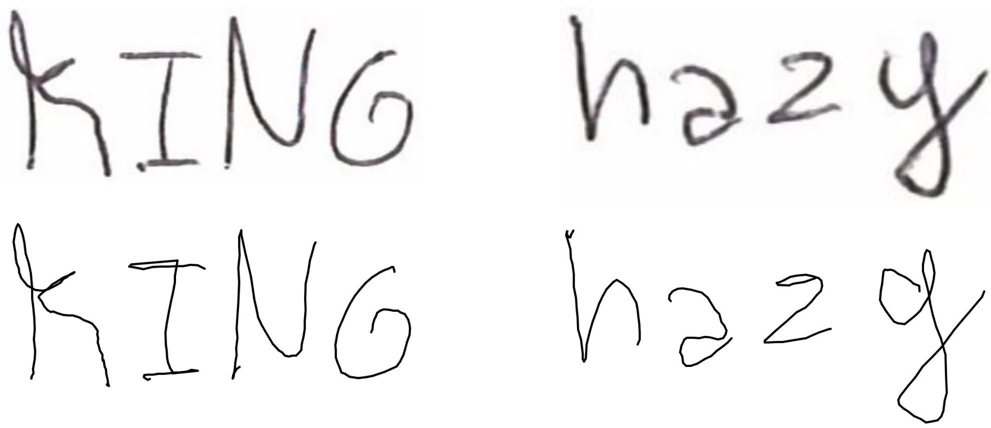
Table 4.1 List of the special symbols and their respective labels for the segmentation process.

Symbol	Label	Symbol	Label
.	dot	[	brackets1
,	comma	]	brackets2
;	semicolon	{	braces1
-	dash	}	braces2
’	apostrophe	"	quotation
:	colon	!	exclamation
(	parentheses1	£	pound
)	parentheses2	%	percent
[	brackets1	?	question

CSV file into n new files, where n represents the number of different labels, i.e., each group of points that forms a unique glyph from the original files. The file’s name is given according to the character and its occurrence number. At the end of this step, we have all glyphs - defined as a set of points - separated in a CSV file.

Figure 4.2 presents a two-word comparison for the offline and online data collected for the subject 04. As we can see in this figure, The Slate may present some problems regarding the sample capturing. That mainly occurs because the points are registered every time the writing tool touches the capture hardware, even if nothing is written down. We can see an example of this issue in the glyphs ‘z’ and ‘y’ from the word ‘hazy’ in Figure 4.2. For some bigger errors, we excluded these points in the segmentation process. However, for small cases such as this one, we purposely did not remove details closer to the letters since it could potentially lose the writer’s characteristics.

Figure 4.2 Offline (top) and online (bottom) data comparison for the words ‘KING’ (left) and ‘hazy’ (right) handwritten from the aubject 04. Some glyphs like ‘z’ and ‘y’ present minor errors from the data acquisition process of The Slate.

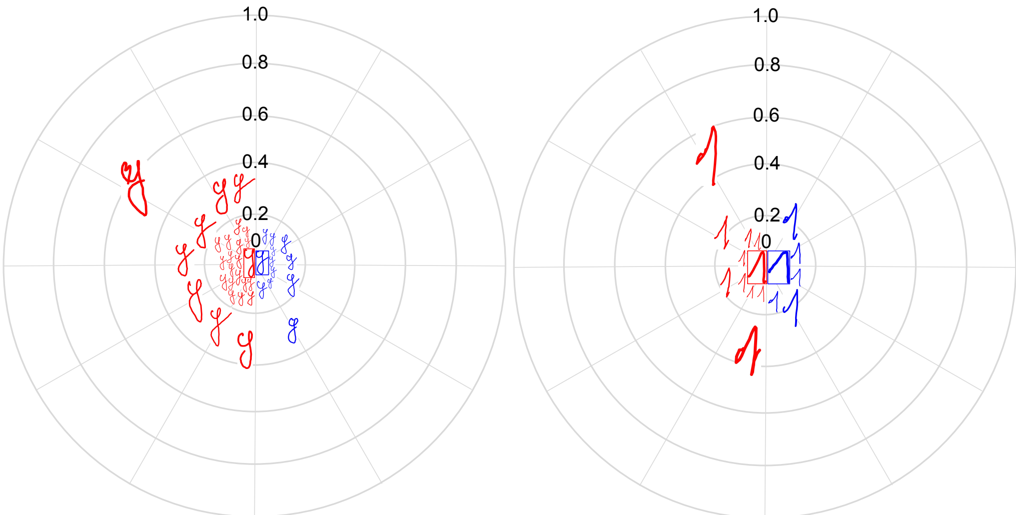


Source: the author.

4.1.3 Shape matching

The goal of this step is to compare the glyphs segmented in the previous step. Given two shapes with one-to-one point correspondences, we compute the distance between two glyphs as the sum of the squared distances between equivalent points, known as the Procrustes method (STEGMANN; GOMEZ, 2002) and usually used in shape analysis. We use the Procrustes return value $d \in [0, 1]$ to measure how similar the shapes are. The closer the distance is to zero, the more similar the shapes are. We opted for this approach since it is simple and deals with eventual translations, rotations, and scales. It also presents a solid alternative compared to more time-consuming techniques involving optimizations (HAINES; AODHA; BROSTOW, 2016) or neural networks (Graves, 2013), which require a larger amount of samples.

Figure 4.3 Character samples *y* and *1* from the subjects 04 and 03. The characters in the center represent the average character and the remaining ones are placed according to their similarity with the average one: the closer their distance to the center, the more similar they are. The red samples correspond to the Universal set and the blue samples to the Pangram set.

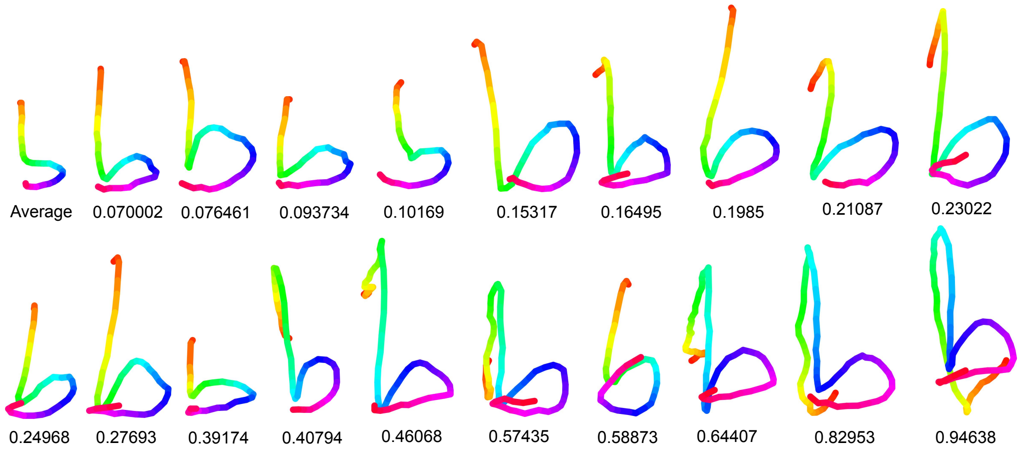


Source: the author.

Since the algorithm requires the same quantity of ordered points to compare, we need to sample both sets such that they have precisely the same amount of points. We used a linear approach that returns points equally spaced along with the points list by downsampling the largest glyph. For instance, if the algorithm is comparing two ‘0’s with 10 and 20 points, the largest glyph reduces its points linearly to contain 10 points, allowing faster Procrustes computation.

Figures 4.3 and 4.4 illustrate some examples on how we can use the Procrustes

Figure 4.4 Example of impact of the Procrustes Distance for the letter 'b' from the Universal text of subject 01. The leftmost upper glyph represents the average 'b', while the other ones represent the Procrustes Distance when compared to the average. From top to bottom and left to right we have the distance values organized in ascending order.



Source: the author.

distance. Figure 4.3 presents all the glyphs for the characters 'y' and '1' from subjects 04 and 03. The glyphs are placed in a circle according to their Procrustes distance when compared to the average glyph. We created the average glyph for each character and subject by expanding the method described above to downsample more than two glyphs at the same time. Once all the same family glyphs have an equal amount of points, we calculate the average position for each point, generating the average glyph. We compute the average as an arithmetic mean, i.e. for each point we sum the x and y coordinates and divide them for the number of glyphs separately.

Figure 4.4 shows the difference for the average glyph 'b' to other original glyphs according to their distances. This figure and Figure 4.5 have their glyphs colored using the HSB color model, varying the Hue value to show the points varying along time. The transition from red to magenta illustrates the trajectory along time of the user's pencil on The Slate.

4.1.4 Data analysis

Since we are checking the similarity between two different samples, we generate two kinds of data tables per subject. The first one compares the glyphs from the Universal set among themselves, while the second one repeats the process for the Pangram set. We compare only the same types of glyphs for both tables; for instance, the lowercase 'a' is compared with all the other 'a' in the same set. We fill the tables with the value d obtained

Figure 4.5 Example of our procrustes triangular matrix. We selected the Pangram set of the character ‘3’ from subject 08, which contains 5 values. We also present in the diagonal the procrustes values for the average glyph against them, showed in the upper left corner.

Avg Glyph '3'	Glyph '3-1'	Glyph '3-2'	Glyph '3-3'	Glyph '3-4'	Glyph '3-5'
Glyph '3-1'	0.0553	0.0684	0.0204	0.0487	0.0172
Glyph '3-2'		0.0639	0.0340	0.0738	0.0410
Glyph '3-3'			0.0688	0.0251	0.0259
Glyph '3-4'				0.0631	0.0594
Glyph '3-5'					0.0142

Source: the author.

with Procrustes. There is no difference in the order of each pair of glyphs compared for both cases, producing a triangular matrix as output, as illustrated in Fig. 4.5.

Our data experiment recorded a total of 79 different glyph types, including 10 digits, 26x2 letters (lower and upper case), and 17 special characters. However, our data analysis focused on digits and lower case letters only since they represent the vast majority of letters in a text. More precisely, investigating results from Jones and Mewhort (JONES; MEWHORT, 2004) regarding the raw case-sensitive single-letter counts from the NYT Corpus, we concluded that the sum of all the lowercase letters occurrences corresponds of around 95% from the total of uppercase and lowercase letters. Thus, we believe that the result obtained by investigating this subset can be easily extended for the upper case and symbols. Although not using them for statistics, we considered them when constructing the Pangram set because subjects are sensitive to frequency relationships among letters (JONES; MEWHORT, 2004), making upper case letters and symbols an important part to preserve written naturalness.

For analysis, at this stage, we generated 72 tables per subject, 36 for the Universal set, and 36 for the pangram. The 36 tables are composed of 26 lower case letters and 10 digits, each one corresponding to the distances from the Procrustes triangular matrices. Next, we condense our information by computing statistical measures (such as median, quartiles, and standard deviation) of d values for the tables. Given the data collected for the Universal and Pangram sets for eight subjects, our hypothesis claims that the smaller subset (Pangram) can produce the same variability as the larger set. The following paragraphs describe the statistical methodology and the data design used to perform the analyses.

To compare the handwriting variability of the samples between the two sets, we perform a non-parametric Wilcoxon Rank Sum Test (abbreviated as WRT) in the summarized d values obtained through Procrustes. More specifically, for each of these 72 tables, for each kind of set, the WRT inputs are the summarized d values given by the median (Md) and standard deviation (sd). For each type of character, including letters and digits, we use the eight subjects as replications. So, for instance, for the character 'a' we have eight Md and sd values for each set and perform the WRT – one for Md and one for sd – to verify if the d values are statistically different between the Pangram and Universal sets.

The H_0 hypothesis states that there is no statistical difference between the two groups against the H_1 hypothesis for the difference. In short, as a previous step for the WRT, for the eight pairs of Md values (one for each kind of set), the internal difference is computed. In the next step, the differences (considering the signal) are ranked. Under H_0, we expected that the sum of ranks due to the negative and positive differences do not differ by an amount. Otherwise, a systematic difference towards one group evidences that the d values are very dissimilar between the two sets. In other words, the amount of variability in reproducing the 'a' character (for instance) strongly differs between the two sets. Thus, if H_0 is not rejected, it means that for the character 'a', there are no significant differences in a handwritten variability captured by the two sets of texts.

Following the aim of this work, we are not interested in rejecting H_0 for any character since we want to show that the Pangram set is enough to capture the handwriting variability. A detailed description of the WRT can be found in (WILCOXON, 1945).

The reason why we are using the two statistics (Md and sd) to compare the two sets of texts is that Md describes the profile value of the d values for each table, while sd describes how much the original d values vary around their profiles. Considering again the eight subjects and the character 'a', if the sd deviations of pangram samples are significantly different from the Universal samples, it decreases the power of the WRT to detect real differences in d values for the two sets for this character by using any profile measure (such as mean and median). We can think that the non-rejection of H_0 for the sd values is a proper validation for the WRT results for comparing groups through the profile measures (Md, in our case). All the WRT runs were evaluated considering the significance of $\alpha = 0.05$.

Figure 4.6 present a graphic summary of the d values (will be commented in the next section) by using Boxplots. They show the general view of the empirical distribution of the data through some order statistics. In short, the d values are partitioned into quartiles, that is, four subsets with equal size. A box is used to indicate the positions of the upper and lower quartiles; the interior of this box indicates the interquartile range, which is the area between the upper and lower quartiles and consists of 50% of the distribution. The horizontal central line of the box represents the Md value of d. The upper and lower dashed lines represent the range of the 25% higher and 25% lower d values. The WRT results and the detailed evaluation approaches are presented in the next section.

4.2 Results

This section presents the comparative study of the handwritten variability captured by the Pangram and the Universal sets. Our validation includes two evaluation approaches: an objective, which measures similarity through the d values from Procrustes and a subjective, that compares visually the two sets by asking a group of users to fill a questionnaire.

In the beginning, we asked 8 subjects to perform the experiment described in Section 4.1. The experiment had average time per subject around 37 minutes and 30 seconds. Also, the average percentage time that the subjects used for writing the Pangram text was around 24% of the total time. This means that the Pangram set took on average $1/3$ of the time it takes to write the Universal set. The information we obtained regarding timing is available in Table 4.2. All our data collected from the subjects, including the original and labeled online data (2D points in csv files) and offline data (digitalized) are available in https://wiki.inf.ufrgs.br/Assessing_Similarity_in_Handwritten_Texts.

Table 4.3 contains the information gathered with the pre and post questionnaires for each subject. Our pre-questionnaire gathered general information about the subjects and their handwriting habits and our post-questionnaire collected data regarding their experience on using the new device. In particular, questions I11 and I13 provided important information regarding the validation of the new hardware when compared to tablet devices. Question I11, which asked The Slate naturalness when compared to pen and paper, received positive feedback with 4.875 average score. Question I13, which compares The

Table 4.2 Information gathered from the experiment regarding time. I1: total clock time in minutes and seconds; I2: total device time in seconds; I3: Universal text device time in seconds; I4: Pangram text device time in seconds.

	I1	I2	I3	I4
S01	42:06	2261.19	1756.88	504.31
S02	40:01	1996.79	1437.04	559.74
S03	40:40	2240.64	1691.76	548.88
S04	40:18	2121.62	1648.27	473.35
S05	36:10	1619.31	1246.91	372.40
S06	40:13	2119.04	1631.02	429.09
S07	33:34	1699.14	1287.32	411.81
S08	26:46	1382.79	1047.93	334.85

Table 4.3 Information gathered from our pre and post questionnaires for our eight subjects. I1: condition (whether it started by writing the Universal or the Pangram set); I2: age; I3: education level; I4: gender; I5: how often do you write by hand? (minutes per week); I6: dominate hand for writing; I7: English native speaker; I8: English reading level; I9: English writing level; I10: English speaking level; I11: how natural is the writing on The Slate compared to common pen and paper?; I12: how experienced are you on using tablet devices? I13: how natural is the writing on The Slate compared to tablet devices?

	S01	S02	S03	S04	S05	S06	S07	S08	Averages
I1	1	2	1	2	1	2	1	2	–
I2	26	21	28	26	23	23	23	26	–
I3	High school	High school	Master’s degree	Master’s degree	Bachelor’s degree	Master’s degree	High school	High school	–
I4	Male	Male	Male	Male	Female	Female	Male	Male	–
I5	15	250	0	90	30	60	30	30	63.125
I6	Always right	Always right	Always right	Always right	Always left	Always right	Always right	Always right	–
I7	Yes	No	No	No	No	No	No	No	–
I8	5	5	5	5	5	3	4	5	4.625
I9	5	4	4	4	4	2	3	5	3.875
I10	5	4	4	3	4	2	3	5	3.75
I11	4	5	5	5	5	5	5	5	4.875
I12	5	4	3	5	5	3	2	5	4
I13	5	5	5	5	5	3	1	5	4.25

Slate comfort usage with tablet devices, also achieved positive score of 4.25. Subjects 06 and 07 which provided average and below average score were also not experienced users with tablet devices.

In the following subsections we describe our two methods for validating our claim regarding the similarity of both texts, which can be summarized in the following hypothesis: "does the handwriting of a person change according to the length of a text?". Following the two main groups of evaluation methods described by Elarian et al. (ELARIAN et al., 2014), we present an objective and a subjective approach.

4.2.1 Objective evaluation

Our main hypothesis to be answered in this subsection can be summarized as follows: "Is the Pangram set statistically equivalent to the Universal set?". In order to validate this claim objectivelly, we worked with our Procrustes data separately for each

Table 4.4 Wilcoxon p-values for Md and sd values for each caracter.

Characters	p-value		Characters	p-value		Characters	p-value		Characters	p-value	
Digits	median	std. dev.	Letters	median	std. dev.	Letters	median	std. dev.	Letters	median	std. dev.
0	0.38228	0.08221	a	0.79845	0.57374	j	0.64538	0.64538	s	0.95913	0.64538
1	0.50536	0.19487	b	1.00000	0.23450	k	0.10490	0.08298	t	0.64538	0.19487
2	0.33566	0.77887	c	0.50536	0.04988	l	0.57374	0.64538	u	0.44180	0.32821
3	0.41359	0.00067	d	0.32821	0.44180	m	0.72090	0.72090	v	0.27863	0.38228
4	0.87848	0.50536	e	0.95913	0.79845	n	0.95913	0.79845	w	0.72090	0.13038
5	0.64538	0.95913	f	0.79845	1.00000	o	0.57374	0.95913	x	0.64538	0.57374
6	0.79845	0.38228	g	1.00000	0.27863	p	0.57374	0.44180	y	0.32821	0.06496
7	0.38228	0.19487	h	0.72090	0.44180	q	0.95913	0.13038	z	0.87848	0.16053
8	0.79845	0.38228	i	0.95913	0.44180	r	0.64538	0.57374			
9	1.00000	0.57374									

character type and subject.

For the two sets of texts, the d values are compared considering: (i) each character separately using the eight subjects as replications; (ii) each type of character (i.e., letters and digits are considered separately) using the eight subjects as replications; (iii) all kinds of characters together using the eight subjects as replications (Fig. 4.6c); and (iv) all kinds of characters together (Fig. 4.6d).

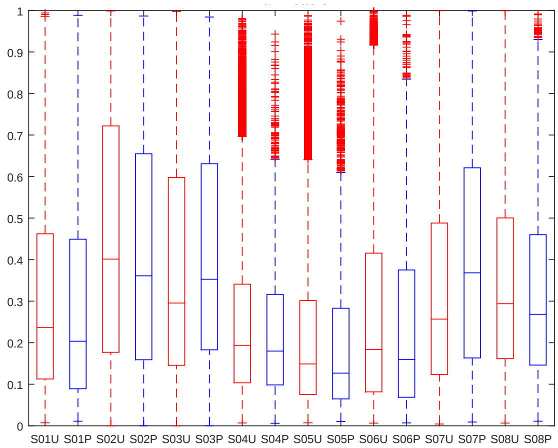
Table 4.4 shows the WRT p-values for Md and sd values for each character. We can see that there is no statistical difference between Md values for the two sets, since $p > 0.05$. The sd values present statistical difference only for the characters c and 3 (significance is quite similar to 0.05 for character c and clearly smaller than 0.05 for digit 3). These overall WRT results reinforce our hypothesis that the pangram seems to be adequate to capture the handwritten variability of subjects.

Figures 4.6a and 4.6b show a Boxplot for the d values for letters and digits, respectively. The red boxes represent the Pangram set and the blue boxes' the Universal set. We can see looking at Figure 4.6a that for nearly all subjects the empirical distributions of d values do not differ strongly between the two texts, except for the subject 07 where Md values are slightly different. Figure 4.6b shows slightly different Md values for subjects 02, 05, 07 and 08. However, by looking at Table 4.5 we can notice that there is no statistical difference between Md and sd values for the two sets. The non significance of WRT results for the sd can be illustrated by looking at the interquartile range for blue and red boxes for each subject in Figures 4.6a and 4.6b. They are quite similar for all subjects.

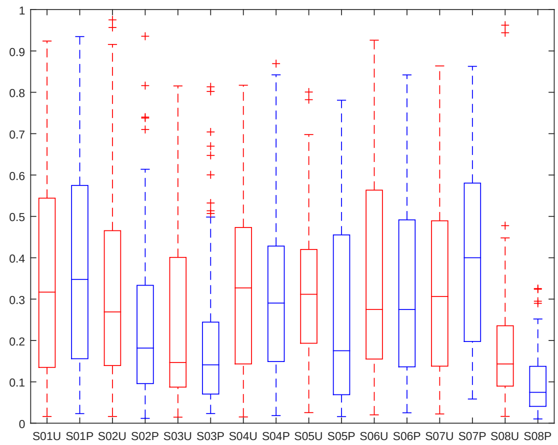
Figure 4.6c shows the Boxplot for the d values considering all characters of any type. We notice again that the empirical distribution of the two texts are very similar for each subject, except for the subject 07 in witch Md values are slightly different again. The WRT results in Table 4.5 for this scenario confirm again the non significance of the differences between the two sets. Finally, by looking at Figure 4.6d we can see that considering all types of characters jointly, there are no strong differences in the distribution of d values between the Pangram and Universal texts. The comparative analysis presented

Figure 4.6 Distances d for the Universal (red box) and the Pangram (blue boxes) of the triangular matrix per subject (a,b,c) and overall (d) according to a specific set.

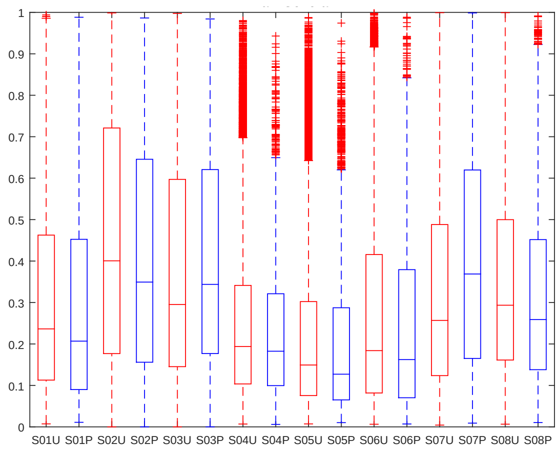
(a) Letters set per subject



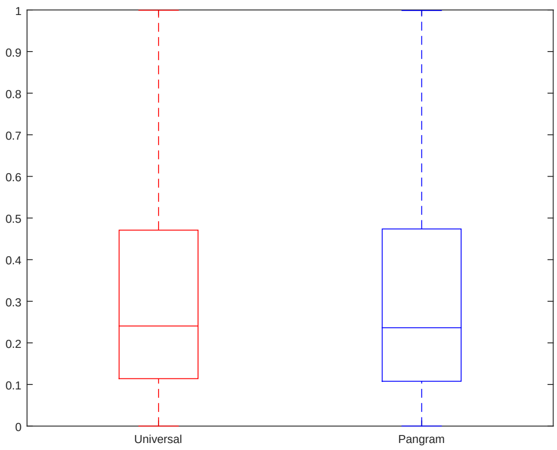
(b) Digits set per subject



(c) General set per subject



(d) General set overall.



by all these boxplots pointed out that the Pangram set seems to be enough to capture the handwritten variability of subjects.

4.2.2 Subjective evaluation

In this section we describe an experiment conducted in order to visually validate our approach by conducting a subjective evaluation method. Our idea is to show that the results from the two texts written by the same person with distinct lengths and at different times are indistinguishable, reinforcing our claim stated in our objective evaluation. Our hypothesis can be summarized in the following sentence: "Is the Pangram set visually equivalent to the Universal set?". In order to achieve this goal, we compared rendered excerpts from the Pangram and Universal sets between themselves, process which we

Table 4.5 Wilcoxon p-values for Md and sd values for the digits, lower letters and both together.

Information	p-value	
	median	std. dev.
Digits and letters	0.5737	0.3823
Only digits	0.9591	0.2345
Only letters	0.7984	0.5054

describe below.

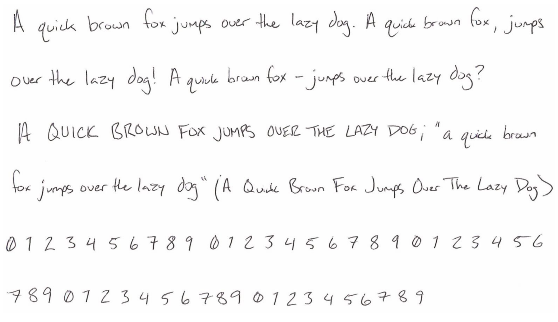
First, we use the raw data obtained from The Slate hardware to create a continuous rendering from the input points gathered from the device using only the information from its 2D coordinates along with the time, i.e., without the label classification. By adjusting two parameters, time threshold and Euclidian distance, we defined a filter to semi-automatically render the text with lines between the points from the original sampling containing the x, y and $time$ coordinates, without previous knowledge from the label information. Since we used the raw values for rendering, we multiply the points by a scale factor in order to scale the whole image to be rendered. Our method draws a line between two points only if the time difference between two adjacent points is less than 0.26 times the scale factor and the Euclidian distance between two adjacent points is less than 1.2 times the scale factor. We achieved these constants throughout experimenting. We compare the multiple rendering excerpts in Figure 4.7.

We constructed a questionnaire based on the one presented by Haines et al (HAINES; AODHA; BROSTOW, 2016), however for another purpose. We opted to present a full rendering approach in contrast to the original digitalized one because we did not want the texture to be a possible feature to compare the words, once we are using the online data approach. We defined 20 questions containing two equal sentences from the same subject, one for each of the two different sets, randomly assigned as A and B along with an excerpt extracted from one of the sets. Each volunteer had the task to select the best fitting set that answers the question by comparing with the sentences above. Figure 4.8 provides one of the questions available from our experiment.

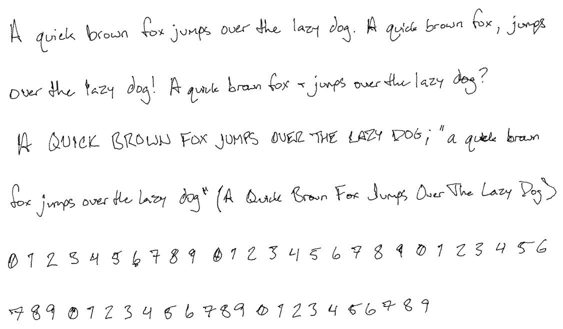
All extracted words were directly rendered from the set of points available from the eight subjects with the same resolution. We organized the questionnaire to contain exactly two questions for each subject, one of them where the excerpt was extracted from the Pangram set and the other one from the Universal set. The excerpts also contained from one to four words. We randomly distributed the 16 questions, organizing them from Q1 to Q20. The remaining four questions consisted of a false positive, which had exactly

Figure 4.7 Rendering excerpt comparison from page 1 of Subject 01.

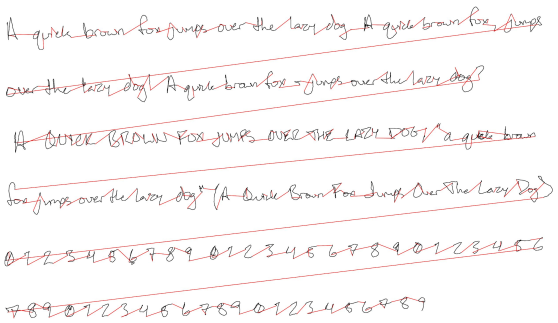
(a) Digitalized page.



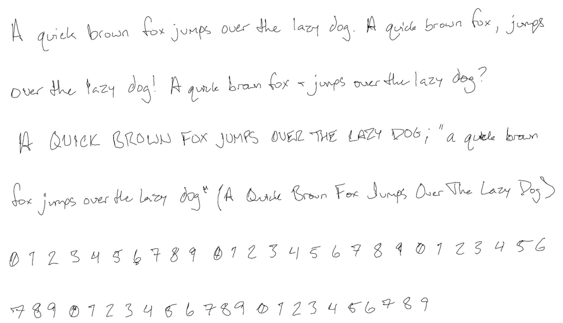
(b) Point-rendered page.



(c) Line rendered page without filter.

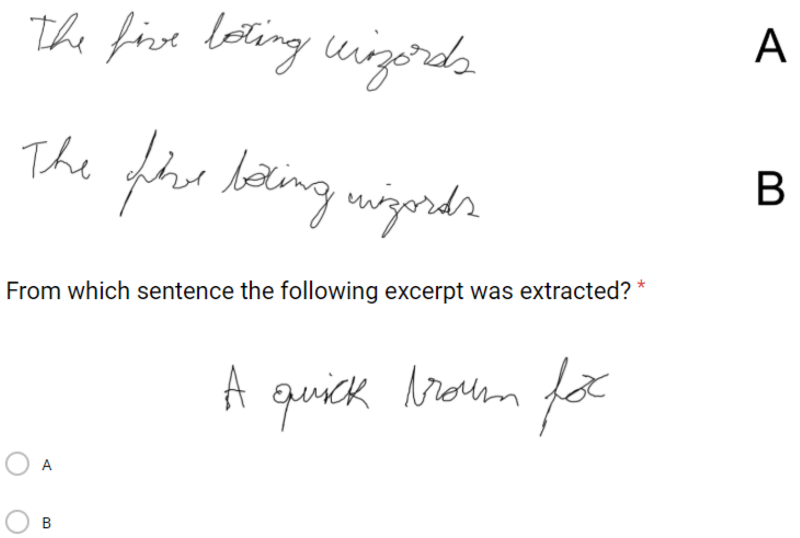


(d) Line rendered page with our filter.



Source: the author.

Figure 4.8 Example one of our questions: on top we present two sentences "The five boxing wizards" attached to a set (A or B). Next we ask the subjects "from which sentence the following excerpt was extracted?", letting the volunteers choose between set A or B.



Source: the author.

the same words from one of the sentences above. We provided these questions only to check whether the volunteers were paying attention to the experiment and we did not compute them in our results. For didactic purposes, we provided an example question which explained the process to the volunteers step by step on what to do, which also we

Table 4.6 Information regarding our experiment questions.

Question	Subject	Correct group	False positive?	Correct letter	Excerpt amount of words
QExample	1	Universal	No	B	1
Q1	8	Pangram	No	A	4
Q2	3	Universal	No	A	2
Q3	2	Pangram	No	A	1
Q4	6	Universal	Yes	B	3
Q5	5	Universal	No	B	3
Q6	4	Pangram	No	B	2
Q7	6	Pangram	No	A	2
Q8	1	Pangram	No	B	1
Q9	2	Pangram	Yes	B	1
Q10	7	Universal	No	A	4
Q11	5	Pangram	No	B	4
Q12	8	Universal	No	B	3
Q13	3	Universal	Yes	A	2
Q14	2	Universal	No	A	1
Q15	4	Universal	No	B	4
Q16	7	Pangram	No	A	3
Q17	3	Pangram	No	A	1
Q18	6	Universal	No	B	3
Q19	7	Pangram	Yes	A	4
Q20	1	Universal	No	B	2

did not count in our result. The complete table containing all the information per question is available at Table 4.6. We also provide the original experiment template, in Portuguese language, in Appendix C.

Our experiment had a total of 175 volunteers, which remotely participated answering our questionnaire. Table 4.7 shows an analysis of the correct answers per subject, per amount of words and per correct set. Only subjects S02, S04 and S08 had a positive hit score, slightly larger than fifty percent. Both excerpts from Subject 02 had only one word, which could explain why it was the easiest subject to discover the correct excerpt on average. Another interesting point emerges when comparing the fragments from the Pangram and Universal sets, making people get more confused when the excerpt was extracted from the Pangram set instead of the Unversal one. Regarding the amount of words, we did not find any correlation. The false positives worked as expected with an almost perfect hit score, showing that people were definitely engaged in the experiment.

Finally, computing all the valid correct answers, we achieved 49.61% hit rate overall. Since the result is slightly less than 50%, it indicates that the subjects do not know which set is the correct one. Thus, we reaffirm visually that the samples written by the same person in distinct times and different text lengths agrees with our objective evaluation, providing no relevant difference.

We also gathered the volunteers's information through a pre and post question-

Table 4.7 Analysis of the correct answers according to the specific groups and general.

Groups	Accept percentage
S01	46.00%
S02	58.57%
S03	40.57%
S04	55.71%
S05	45.71%
S06	48.57%
S07	48.00%
S08	53.71%
Universal	53.79%
Pangram	45.43%
1 word	44.86%
2 words	63.29%
3 words	52.71%
4 words	37.57%
False positives	98%
General	49.61%

naires. We built our questionnaire in Portuguese, since we wanted to achieve a larger public by publishing it not only in our Institute, but also in the social networks. Differently from the writing experiment, we believe that good fluency in English was not a prerequisite for completing this experiment, since both languages have the same script to help comparing the characters.

Our pre-questionnaire gathered general information about the volunteers and their handwriting habits and our post-questionnaire collected data regarding their experience on answering the questions. The questions asked are the following: I1 - age; I2 - education level; I3 - gender; I4 - what device (desktop or mobile) are you using to answer the questionnaire?; I5 - do you have any visual issue? which one? I6 - have you previously participated in an experiment from the same authors before? I7: is English your mother language? I8: what is your English reading level? I9: what is your English writing level? I10: what is your English speaking level? I11: how hard do you classify this task?; I12: which strategy did you use to differentiate the two sets? I13: Please provide any free commentaries regarding the experiment in general if you desire so.

Particularly the question I11 shows that the task result was between medium and hard, achieving an average score of 3.502. Regarding question I12, 98.9% of the volunteers selected the shape of the characters/words option, followed by 40.6% for the pairing of the characters/words and 33.1% for the spacing among the characters/words.

4.2.3 Implementation details

Our project was developed in C++, Matlab and Processing. We used C++ for the steps defined in subsections 4.1.1 to 4.1.2. We worked with the API provided by ISKN and we used the chrono C++ library to measure the time stamps. For the subsection 4.1.3 we used Matlab since it provides effective ways to deal with large data besides containing an implementation of the Procrustes and Wilcoxon rank sum function. We applied the Wilcoxon rank sum test according to the function ranksum available in MATLAB (GIBBONS; CHAKRABORTI, 2011) (HOLLANDER; WOLFE; CHICKEN, 2013). We also used the Processing language to render the glyphs and the handwriting samples. All the questionnaires we provided were created using the Google Forms.

5 HANDWRITING SYNTHESIS FROM PUBLIC FONTS

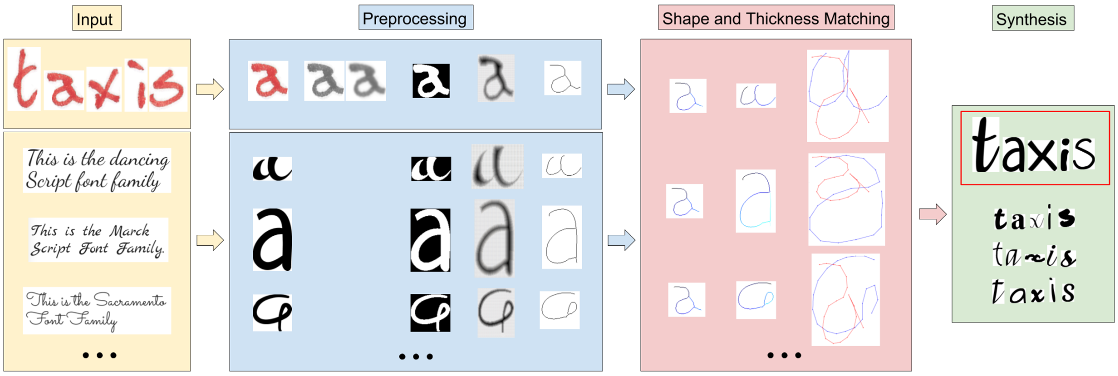
The goal of handwritten text synthesis techniques is to provide images that look like they were written by a human hand, either by an unidentified author or by a particular person, which constitutes the harder problem. In this chapter we discuss a novel approach for handwriting synthesis, aiming to reduce the computational cost when compared to state-of-the-art techniques while still providing a natural output. We also focus on using low variability sampling data, which is supported by the results discussed in Chapter 4. Our technique focuses on the user-specific character matching using a mathematical technique called Procrustes Distance, which can be used to compare two sets of points. In this chapter we present our offline approach, which works with public fonts to increase the variability from the input sampling.

From an input sample of the desired handwriting, we introduce an algorithm that finds the best match between characters using as source for the output text the large collection of publicly available fonts designed to look like handwriting. For each character in the desired output text, we find the best match among the public fonts using a metric that matches both the shape and appearance of the input real character. Once we have the set of best characters we build the output sentence or paragraph by concatenation of individual characters.

Although human writing is individual and can be traced to a particular person (SRIHARI et al., 2002), we claim that the large number of fonts used as input for synthesis provides a degree of variability which allows renderings that look like were handwritten by a particular person, without the expensive cost of previous approaches. Our central insight is the use of a large number of publicly available fonts to find the best match to a particular person's handwriting. This approach can also be categorized as a beautification one, since transforms the original calligraphy to a computerized one following some pattern.

This chapter is divided into three sections: we first describe our methodology, which is subdivided in further four steps. Then we present our results and, in the final section, a discussion regarding this approach.

Figure 5.1 Overview of our technique. Input: sample of handwritten text and families of handwritten-like fonts. Preprocessing: process the images representing the glyphs to extract information needed for the shape matching step. Shape and thickness matching: compares each glyph from the user input with all the font dataset glyphs in order to find the best match. Synthesis: concatenate the best glyphs to simulate the text input positions.



Source: the author.

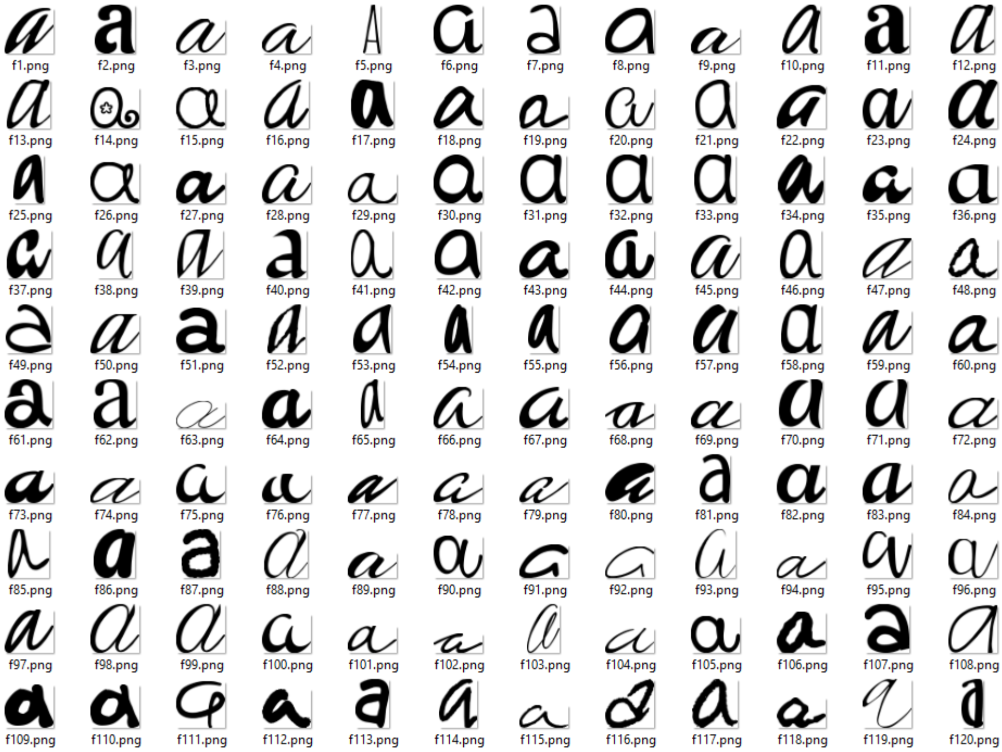
5.1 Methodology

In this section, we present our technique for handwriting synthesis using public fonts. Given a handwritten text and families of fonts as input, our aim is to match, for each real input glyph, the best one from the families of fonts. Once we have the collection of best matches, we can render any string similarly to the real input handwriting. Our inputs are the sample of handwritten text that we want to replicate, plus a collection of families of fonts that are publicly available. There are many possible options to capture a person's written text to use as input. Input samples captured with pen or pencil on paper are sometimes preferable to use as input since this mode better captures hand flow, but need to be digitized later. Samples collected from direct writing on tablets are already digitized but are usually less natural. Therefore, we used as input some of the samples written on paper available from (HAINES; AODHA; BROSTOW, 2016), among others.

As our second input and source for the public fonts, we collected 120 families of fonts that are carefully designed to be similar to handwriting. Of these, 87 families are available as Google fonts (<https://fonts.google.com/>) and the remaining 33 families are available from the web portal Free Calligraphy Wedding (34..., 2017) (18 families) and from the web portal 1001 fonts (1001..., 2017) (15 families). From this last portal, we used the families arbitrarily selected from the most popular ones. Fig. 5.2 (Input) illustrates three of the Google families and Fig. 5.2 shows the glyph 'a' for all 120 families.

Given a sample of handwritten text, we first manually segment each glyph from the input sentence, generating an image for each. Then, we execute the next steps in order

Figure 5.2 glyph ‘a’ for all 120 families of fonts used as input.



Source: the author.

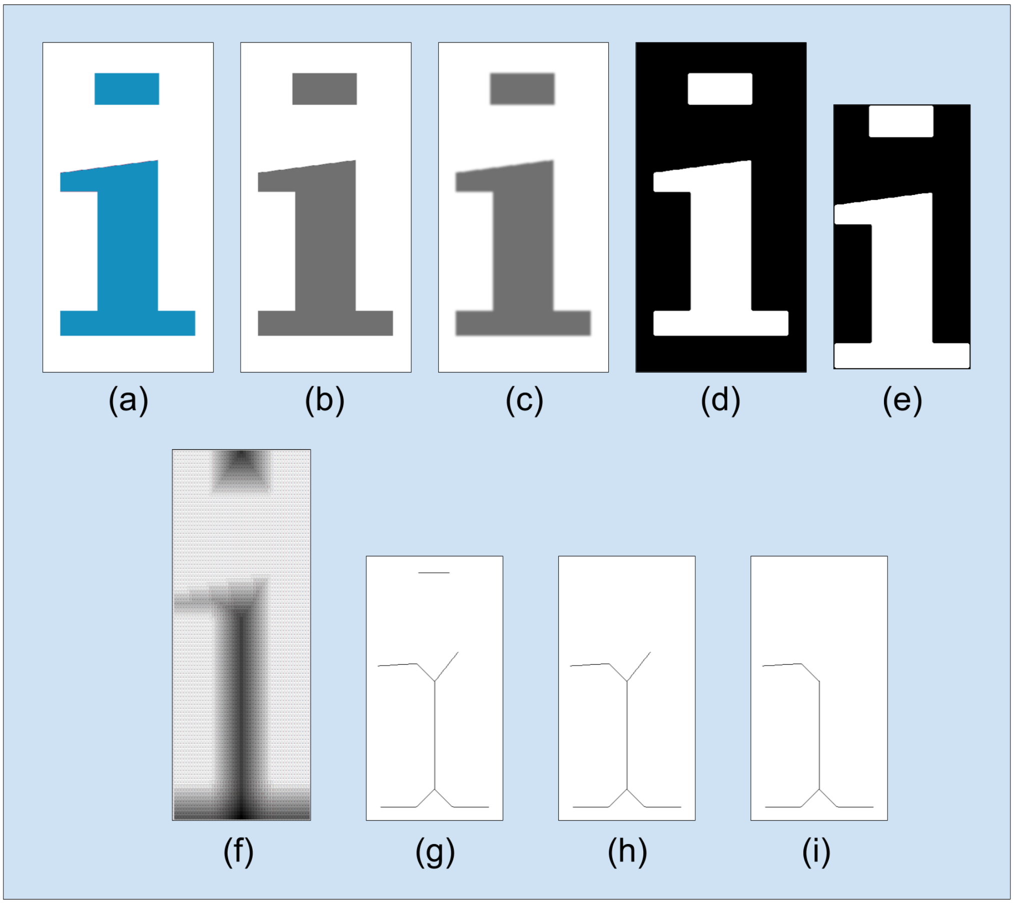
to select the best glyph among those of same type from the font families. For instance, an ‘a’ glyph will only be compared with other a’s. Our main technique is divided into three steps: preprocessing, shape matching, and synthesis, detailed below. The synthesis part is user-assisted whereas the shape matching and preprocessing steps are automated, although some parameters can still be calibrated to control the results. In Fig. 5.1 we present the overall process.

5.1.1 Preprocessing

Our preprocessing performs operations on the inputs to extract the necessary characteristics for the comparison described in Section 5.1.2. We apply simple image processing operations on the inputs, followed by a specific thinning strategy to capture specific features from the images. Fig. 5.3 illustrates all preprocessing operations using a simple drawing of an ‘i’ glyph that has some specific characteristics and triggers all the six steps described below.

1. Basic operations on the sample input: since we do not use the pen texture, we start by converting the color image into a gray level one. Digitized natural handwritten

Figure 5.3 Preprocessing pipeline for the glyph ‘i’ . (a) original image; (b) image a) converted to gray level; (c) image b) with gaussian filter; (d) image c) binarized; (e) image d) with bounding box + 1 pixel; (f) distance transform applied to image e); (g) image e) after thinning; (h) image g) after disconnected paths removal; (i) image h) after the small ramifications removal.



Source: the author.

text presents small variations intrinsic to the writing process which appear as noise at the border of the characters. We apply a Gaussian filter to remove this noise, which helps compute the glyph skeleton in the next step. We also binarize the image according to a threshold to separate the background (black) from the foreground (white). Next, we crop each image to adapt to its bounding box plus 1 pixel in each dimension to avoid future problems with the thinning algorithm. Fig. 5.3-a to 5.3-e illustrate these steps.

2. Size normalization: in this step we generate, for each glyph from each font family, an image that matches approximately in size the resolution of the corresponding sample input. Since the designed glyphs have very different aspect ratios, we need a criterion for normalization. For instance, if three occurrences of glyph ‘a’ are found in an input text, and the largest one is 90 pixels wide, then glyph ‘a’ for all font families will have this same width but different heights, according to its original design and maintaining

its aspect ratio. In this step, we also repeat all the basic operations described in step (1) above for the resized font images, except the Gaussian filter. Considering that the fonts usually do not have noise, we thought that it could erroneously mask some intended artistic effects created by the font authors.

3. Distance transform: in this step, we compute the distance transform (FABBRI et al., 2008) for both the input text and the font images. We use it later in our thickness metric. Fig. 5.3-f shows a representation of the distance transform: the bigger the numbers, the stronger (blacker) the pixels.

4. Skeletonization: we apply the thinning approach by Zhang and Suen (ZHANG; SUEN, 1984) to compute the skeleton of the glyphs. The images have the same size as described in step 2. After computing the skeletons, we still need to perform two operations on the pixels describing the skeleton preparing them for the application of our distance metric. First, we convert the skeleton into a one-pixel wide one through the algorithm described in (LAM; LEE; SUEN, 1992). This is important because our distance metric needs a continuous set of points, which is detailed in the next subsection. Second, we compute the skeleton's *end* and *branch* pixels, which will have an important role in the next steps. End pixels are all the pixels that have only one neighbor, whereas branch pixels are all the pixels that have three or more neighbors, both considering an 8-connected neighborhood. Fig. 5.3-g presents the skeletonization obtained from Fig. 5.3-e.

5. Disconnected paths removal: this operation consists in removing all the disconnected pixel paths except for the largest one, which will be considered the main skeleton. We start from an endpoint and count the number of pixels found, erasing them until they have all been processed. If an image has no end points, we start with the bottom-most left pixel of the skeleton by default. When a branch pixel is found, we store it for a later check when a path is over. When a path is over, we store its number of pixels and start all over again while the number of visited pixels is different from the total of the skeleton pixels. The longest path is preserved as an image, and the others are erased. Fig. 5.3-h illustrates this step. The algorithm finds two disconnected segments and removes the upper line which had fewer pixels than the body of the 'i' glyph. As a side note, in English the majority of handwritten characters do not have disconnected parts, since English does not have accents.

6. Small ramifications removal: here we remove small pixel ramifications that could be considered as noise produced by the thinning algorithm. A ramification occurs when a pixel path starts on an end pixel and goes through a branch pixel or vice-versa. We

defined as noise ramification all ramifications that have at least 10% of the total number of skeleton pixels, not counting branch points. We chose this percentage through simple experiments. Then, all noise ramifications are removed from the path. These paths are easily detected using a variation of the algorithm described in the last step. Fig. 5.3-i shows the upper right ramification removal, which in this case helps to improve the skeleton result considering the shape of glyph ‘i’. Although it produces good results for several cases, we cannot assure the ramification is not an important stroke removed from the character.

We now have the final skeleton of all images to compare each glyph from the input text with the glyphs from the fonts. The next subsection describes the algorithm to compare shape and thickness among glyphs.

5.1.2 Shape and thickness matching

For each glyph in the input text we have to find, among all available public fonts, the best match. For this task we define a metric that first compares the glyph shapes followed by a comparison of their line thickness. Our shape metric requires as input one-to-one point correspondences between the two shapes. This means finding an ordered list with the same amount of points for both glyph skeletons. We compute this list from the pixels that define the skeletons. We assume all the preprocessing steps are already done at the beginning of this stage. We divided this process into five steps, detailed below.

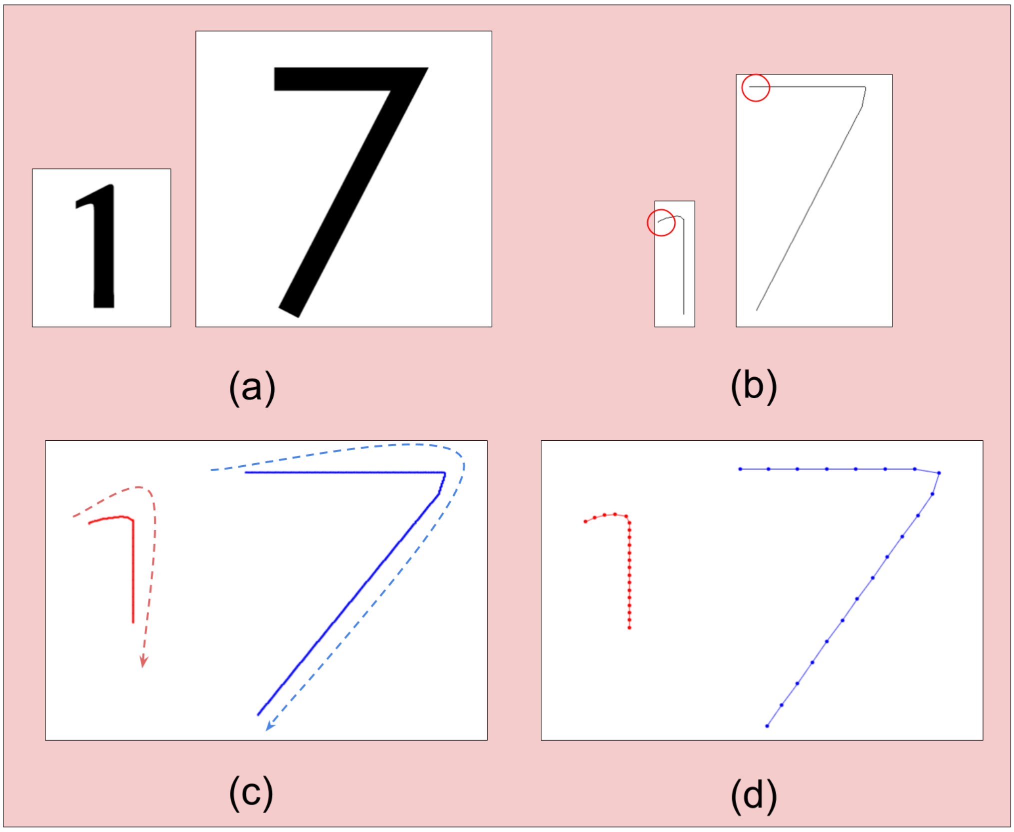
1. Defining the starting pixels: since we will create an ordered list of pixels for each skeleton, our first task is to choose the most similar starting pixel for both skeletons. We accomplish this by using a simple Euclidian distance through all the end pixels respecting the size normalization of both images. As described in Section 5.1.1 step 5, if an image has no end pixels, we start with the bottom-most left pixel by default. The pair chosen is the one that has the least distance found among all the comparisons. Fig. 5.4-b illustrates the starting point found for both example images in 5.4-a.

2. Ordered pixel list creation: starting from the initial pixels found in the previous step, we need to go through both skeletons capturing all pixels along the way. For each pixel from the starting one, we decide the next 8-neighbor according to the following priority: branch point, then clockwise order starting from the bottom pixel, as illustrated in Fig. 5.5 for a sequence of pixels. We mark each visited pixel and every time a pixel has no new neighbors we backtrack through all the pixels already visited, beginning with

the branch points in their visited order until there is no one left. Since all the pixels are guaranteed to be connected because of our preprocessing phase (step 5), the algorithm will surely end at some pixel. The use of this simple rule makes sure we are moving through the same path when creating the list of pixels for both skeletons. Not only that, but we also guarantee a consistent path even for considerably different skeletons. Fig. 5.4-c shows the list of pixels and their direction on the list. At the end of this step, we have an ordered list of pixels for each skeleton.

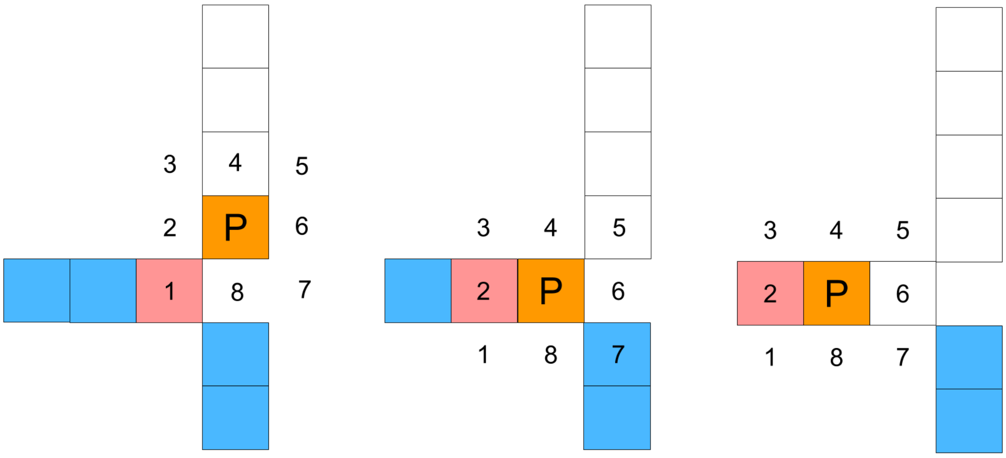
3. Point list creation: since the shape matching requires two sets of the same quantity of ordered points, we need to sample both sets of ordered pixels such that they have exactly the same amount of points. We used a linear approach which given the number of points to be sampled returns points equally spaced along the pixel list. If one ordered list of pixels has fewer points than asked, we sample everything possible and just repeat the last value till it reaches the given number. Even though this parameter can be changed, we opted to fix it to 20 points through all the experiments, since it presented

Figure 5.4 Pipeline for computing an ordered pixel list for the skeletons of two characters representing numbers. (a) input images of a '1' and a '7'. (b) result of the preprocessing phase for both images in a) and their respective starting pixels selected. (c) ordered list of pixels and their direction on the list. (d) points from c) after linear sampling for 20 points.



Source: the author.

Figure 5.5 Ordered pixel list creation. P represents the current pixel and the numbers 1 to 8 express the priority to visit the next pixel. Example of three iterations to show the pixel to be chosen. The white pixels are already visited, the blue ones still need to be visited and the red pixel is chosen to be visited next.



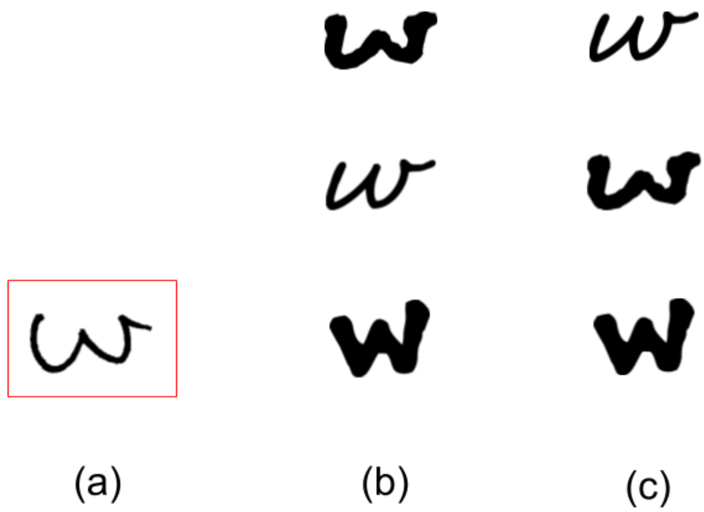
Source: the author.

good results overall. Fig. 5.4-d illustrates the final points chosen.

4. Shape matching: the shape comparison uses the Procrustes distance applied to the list of points obtained in the previous step. Given two shapes with one-to-one point correspondences, we compute the distance between them as the sum of the squared distances between equivalent points, known as Procrustes method (STEGMANN; GOMEZ, 2002) and commonly used in shape analysis. We used Procrustes since it is simple and deals with eventual translations, rotations and scales. We use the return value $d \in [0, 1]$ to measure how similar the shapes are. The closer to zero, the more similar the shapes are. On a small subset of the best matches found using the Procrustes distance, we apply a second metric that compares the thickness of the characters, described in the next step.

5. Thickness matching: our line thickness comparison is processed independently of the shape matching and consists in comparing the points using the distance transform previously calculated. For each sampled point on the list, we get its value in the distance transform matrix for both images. We calculate the absolute difference for each pair of values and keep on summing them all until no pair is left. At the end we obtain a value $t \geq 0$ representing the thickness. Just as with the distance metric, the closer to zero, the better the result. Since the points have a one-to-one correspondence, our approach captures the difference of thickness between the characters. In Fig. 5.6 we illustrate a glyph and the results of first applying the shape metric followed by the thickness metric. We can see the reordering of the best matches according to the thickness criterion.

Figure 5.6 Shape and thickness results. (a) glyph ‘w’ handwritten. (b) best three ‘w’ computed using only the shape comparison. (c) best three ‘w’ of our shape metric ordered by the thickness algorithm.



Source: the author.

5.1.3 Synthesis

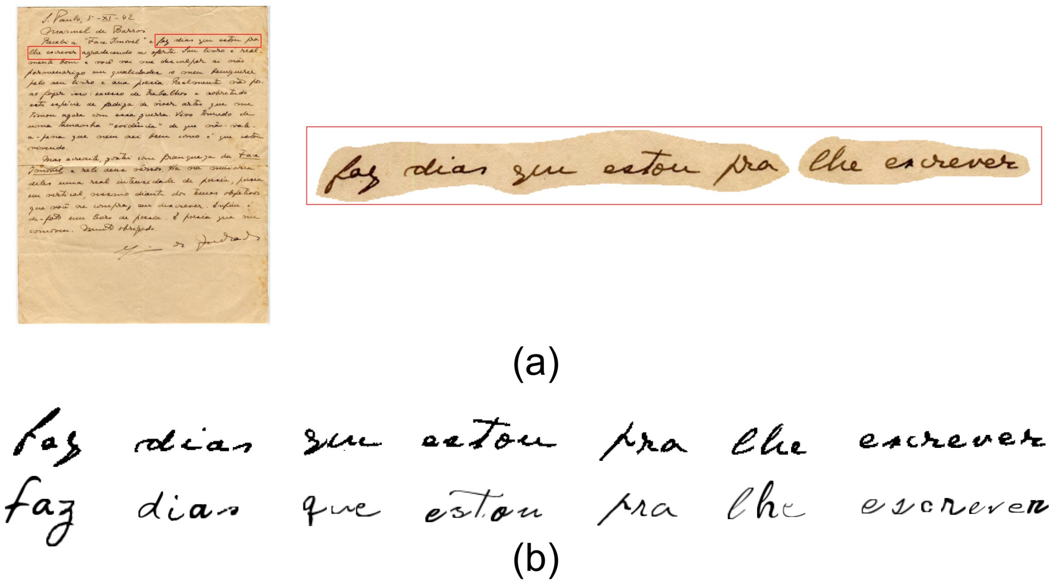
This is the last and simplest step. Once we have the glyphs that minimize the distance between the input and the available public fonts, we are ready to write the output as a collection of glyphs that will be placed side by side such that they look like they were written as a unique text. When available, we tried to manually mimic the same vertical and horizontal spacing of the original whenever possible.

5.2 Results

Here we present the main results of our method. We implemented our prototype using Processing and Matlab. We computed our results in an i5 3.50GHz. First, we present examples of possible uses for our results, followed by the results we collected from a user study.

In Fig. 1 we show a synthesis result from an excerpt of a letter from Mário de Andrade, an important Brazilian writer. We can see that even without the ligatures among glyphs typical of cursive writing, the overall result is still similar to the input. This result took 306 milliseconds in total to compute: $146ms$ for generating all the font images for all the glyphs and $160ms$ for computing the matching among glyphs. The largest bounding box of all the input was for glyph ‘f’ with 42 x 43 pixels.

Figure 5.7 Result of our handwritten synthesis approach. (a) Digitized letter from Brazilian writer Mario de Andrade and excerpt of the letter used as input to our algorithm. (b) On the top the same excerpt processed to remove texture and color and below our synthesis result.



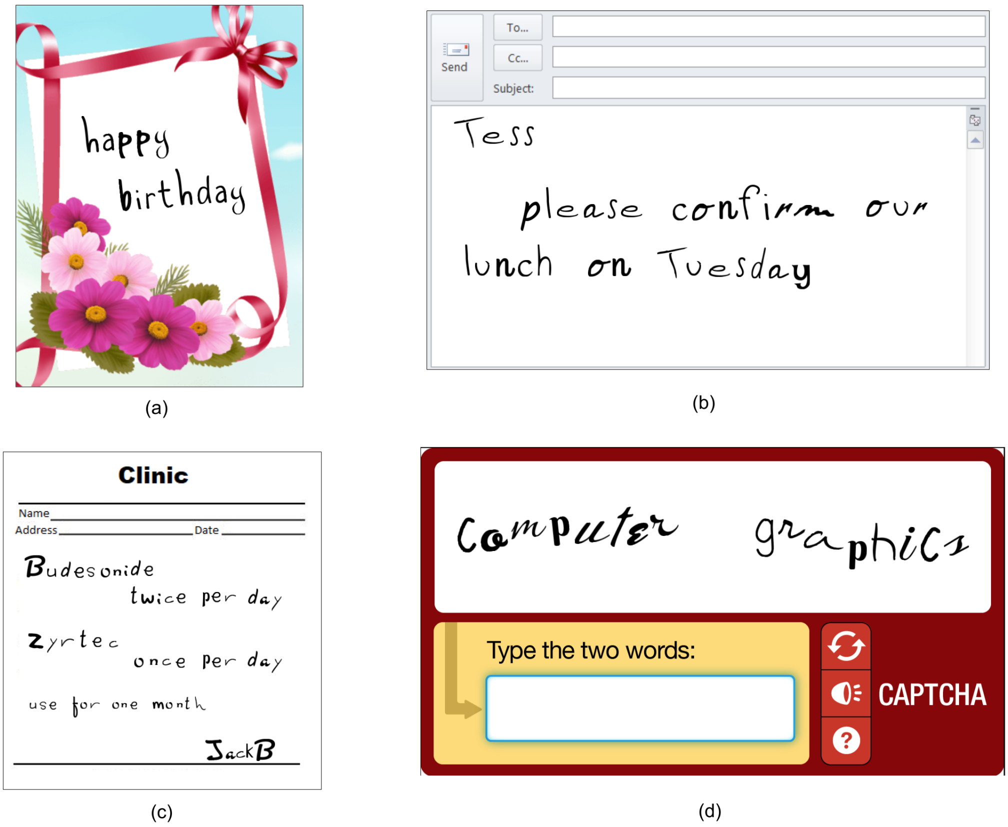
Source: the author.

We also present other examples of applications of our approach in different contexts. For input, we used the sample handwriting from subjects 2 (S02), 3 (S03), 4 (S04), 6 (S06) and 9 (S09) from (HAINES; AODHA; BROSTOW, 2016). These samples provide variability among individuals as well as variability with respect to the sentences and writing tool used. S02, S03 and S06 used a fine liner pen; S04 used a fountain pen, and finally, S09 used a gel pen. Fig. 5.8 (a) presents a message created using the best matched fonts for S02 in a flower card, (b) S03 writing in an e-mail, (c) S06 in a medical prescription, and (d) an artificial CAPTCHA. For the artificial CAPTCHA we used the glyphs combination that was considered the worst in the study we performed to assess our results, described next.

We compared our results directly with (HAINES; AODHA; BROSTOW, 2016) through a paragraph written in Sir Arthur Conan Doyle’s real handwriting. We synthesized the sentence ”Elementary my dear Watson”, presented in Fig. 5.9. We extracted the samples from the same paragraph used in (HAINES; AODHA; BROSTOW, 2016), although in our case we do not have individual variation for glyphs of the same letter. Even though our solution is simpler than (HAINES; AODHA; BROSTOW, 2016), our synthesis still provides satisfactory results for many applications that do not demand full similarity. Further, we compared our results with the website WhatTheFont (WHATTHE-

FONT, 2017) in Fig. 5.10. We used as input a sentence from user S09. The website generated the top 10 best matches. We selected the first of each and placed them side by side. Even though a few characters are visually similar, in general, for real handwriting samples, WhatTheFont has difficulties in providing adequate suggestions.

Figure 5.8 Examples of synthesized applications using the best match from three different subjects from (HAINES; AODHA; BROSTOW, 2016) and the 100th match from the first subject. (a) birthday card (S02). (b) E-mail (S03). (c) medical prescription (S06). (d) artificial CAPTCHA (S02).



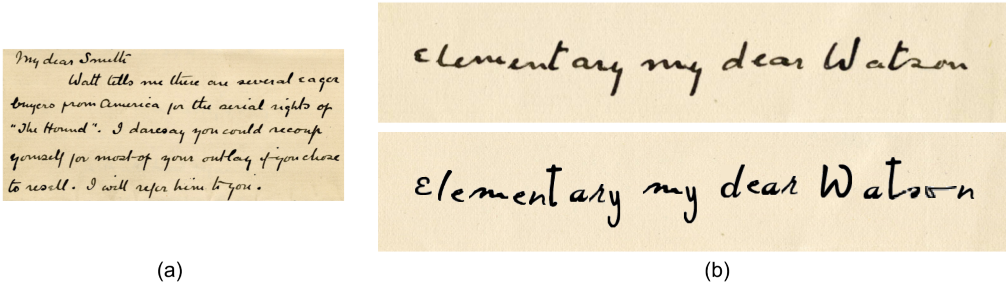
Source: the author.

5.2.1 Validation

We designed an experiment to assess how similar to the human input our results are. For input, we used the same as before, that is, the handwriting from S02, S03, S04, S06 and S09 from (HAINES; AODHA; BROSTOW, 2016). Fig. 5.11 shows four out of five results we obtained by running our algorithm. The original sentences are highlighted in a red box and our results highlighted in blue.

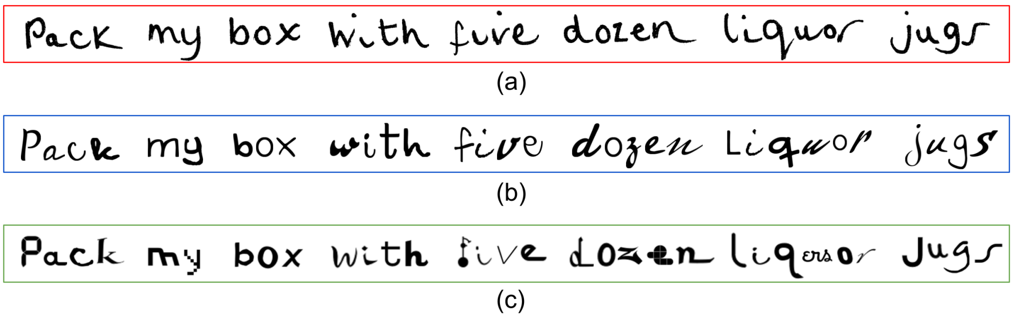
Since handwriting is usually seen on paper, we provided printed copies of the real handwriting input, printed at the top of the page, together with 5 of our results randomly

Figure 5.9 Comparison between results from (HAINES; AODHA; BROSTOW, 2016) and ours for a letter wrote by Sir Arthur Connan Doyle. (a) Sampled paragraph of the original letter. (b) Result synthesized by (HAINES; AODHA; BROSTOW, 2016) on the top and our synthesis result below.



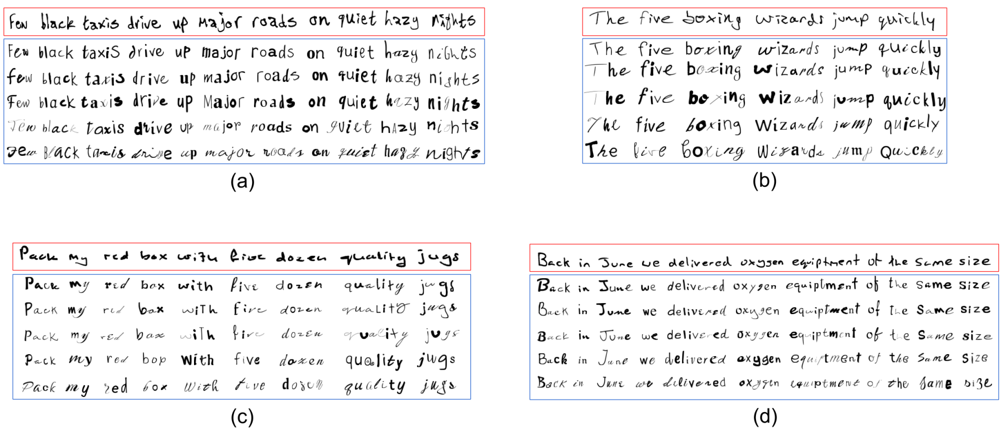
Source: the author.

Figure 5.10 Comparison between results from the website WhatTheFont (WHATTHEFONT, 2017) and ours using a sample of S09 (HAINES; AODHA; BROSTOW, 2016). (a) Original sample. (b) Our best result. (c) WhatTheFont first result.



Source: the author.

Figure 5.11 Four out of five synthesis results used in our experiment. All the sentences are ordered from top to bottom: original (highlighted in red), 1st, 2nd, 3rd, 40th and 100th (highlighted in blue). (a) S02. (b) S03. (c) S04. (d) S06.



Source: the author.

positioned below the original handwriting. We ordered the results initially by the shape metric and then, only for the first three best matches, we ordered again by using the thickness metric. We provided 3 of our best matches along with the “best” results in positions 40th and 100th. We asked the subjects to grade each result on a scale of 0 (zero) to 10 (ten) assessing how similar they considered each result when compared to the original handwriting. Twelve subjects took part in the experiment, with an average age of 25 years, all Computer Science students. We had two different test versions in which we shuffled the answers and the subjects order. For the original template with ordered answers of our subjective evaluation, please access our Appendix D.

Table 5.1 presents the average of grades for all cases. For the first three subjects, the results of our technique received the highest grades, as expected. For all subjects, the best grades were consistently assigned for the best three synthesized sentences. Further, we obtained an overall average grade of 7.1 considering only the highest scores for each result.

Table 5.1 Results containing the average scores of our experiment. In bold, the highest grades for each subject.

	S02	S03	S04	S06	S09
1st sentence (best)	**6.42**	**8.17**	**7.33**	5.58	6.33
2nd sentence	**6.42**	6.33	6,25	**6.75**	6.75
3rd sentence	5.17	5,17	4.42	5.00	**6.92**
40th sentence	3.08	3,42	2.75	3.58	4.25
100th sentence	1.33	2.08	3.17	2.58	1.83

6 CONCLUSIONS

This thesis presented a few contributions to the handwriting computing field. This chapter starts by summarizing our achievements and presenting our limitations and future works.

First we have conducted a study to answer whether the amount of collected data from handwritten texts has an impact on the user's character variability. We presented two evaluations, covering both objective and subjective approaches. Our results show that the samples collected with our specific text using pangrams are statistically equivalent in variation with samples collected using general texts for lower case letters and digits. This finding allows handwriting techniques to reduce their sampling texts, requiring a shorter time to gather handwriting information from subjects. We also provide an online public dataset containing the information gathered from eight subjects using a tablet-like device that allows users to write using common pen and paper. This special device allowed us to obtain online data without the well-known distortion provided by common tablets.

Next we have presented a technique for the synthesis of text that looks like it has been handwritten according to a particular style from a person. Given a user text sample and some families of fonts, our approach finds the fonts that best match the user calligraphy. Each person may have their own font family created by the concatenation of the best samples from several families. This can also be considered as a beautification technique by replacing the original calligraphy to a computerized one. We accomplish this by a preprocessing step followed by a matching and thickness step, applied to individual characters. We tested our approach with low resolution glyphs extracted from a real letter and achieved visually similar results. Besides, we conducted a user validation study that presented positive results overall. Our technique has several uses, which we demonstrated through some simple examples from artistic applications to CAPTCHA generation.

6.1 Limitations

Regarding our handwriting variability study, our technique may suit only approaches that do not work specifically with a large amount of original characters, like the one with neural networks, for instance. We also collected and classified the upper case letters and special symbols, but did not consider them statistically because we did not want to increase the Pangram set with less important letters. We believe, however, they would provide the same results we obtained for the lower case.

Our synthesis technique also has a few limitations. Both the text segmentation and the glyph concatenation uses manual work. This requires some effort to position the glyphs correctly so that they resemble the original. The thinning strategy can ignore some important features of the glyphs and the branch removal sometimes may delete an important part of a letter. Also, besides very simple and well suited for handwriting, the Procrustes method is not as powerful as some other techniques that involve neural networks and optimization algorithms overall, which can produce more robust results. We also provided a pure concatenation technique, without dealing with ligatures.

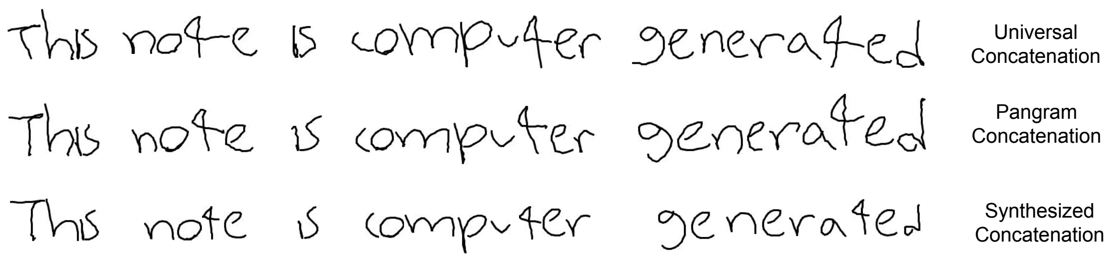
6.2 Future work

Our next steps towards the handwriting analysis include a plan to develop a synthesis algorithm using the Pangram text set coupled with statistical handwriting model that synthesizes various characters while preserving a writer's style (CHANG; SHIN, 2012). Chang and Shin (2012) results showed that many characters can be synthesized from a limited amount of data, exactly our case with the Pangram set. More ambitious goals are to invest in a method for semi-automatic segmentation and to expand the studied characters to other scripts. We also want to increase our database, allowing the application of neural networks techniques shape comparison besides Procrustes. In order to increase our validation methods, we intend to work with graphologists.

For future work regarding our synthesis technique, we aim to work towards the limitations and expand our inputs to deal with accents and other languages besides English. Furthermore, we intend to explore new matching techniques, such as Iterative Closest Point and Hausdorff Distance. It would be very nice to also replicate the texture of the original handwriting, when available, as done in (HAINES; AODHA; BROSTOW, 2016). We intend to increase the number of families of fonts, support online input handwriting connecting with our handwriting analysis and synthesize the ligatures among glyphs of different families of fonts, increasing the visual similarity. Ligatures could be incorporated to connect the chosen font glyphs according to the information extracted from the user.

We also present a prototype representing the possibilities for future work in the online synthesis area. We present a concatenation approach constructed from the average glyphs of the Pangram set. For each character in the target text, i.e., the one we want to synthesize, we start by downsampling their points according to the glyph of the character set that has the least amount of points. Then, we simply compute the average of all these new points, producing a new character. Once we have all the average characters, we construct the sentence by manually concatenating the synthesized glyphs. A sample of this technique for the Subject 05 is presented in Figure 6.1, producing the new sentence "This note is computer generated". We also compared the new synthesized sentence with a simple glyph selection from the Universal and Pangram texts. Plus, in order to produce different results for the same character occurrences, we changed the input set by removing some of the glyphs, thus producing a new average one. Given the similarity of the results achieved with our prototype, we intend to further explore this approach.

Figure 6.1 Comparison of our average online synthesis technique with selected Universal and Pangram glyphs from Subject 05.



Source: the author.

REFERENCES

1001 Fonts. 2017. <http://www.1001fonts.com/handwritten+handwriting-fonts.html>. Accessed: 2018-04-14.

34 Free Calligraphy Script Fonts for Wedding Invitations. 2017. <http://www.prettyweddingpaper.com/34-free-calligraphy-script-fonts-for-wedding-invitations>. Accessed: 2018-04-14.

ABED, H. E.; MÄRGNER, V.; KHERALLAH, M.; ALIMI, A. M. Icdar 2009 online arabic handwriting recognition competition. In: IEEE. **2009 10th International Conference on Document Analysis and Recognition**. [S.l.], 2009. p. 1388–1392.

AKSAN, E.; PECE, F.; HILLIGES, O. Deepwriting: Making digital ink editable via deep generative modeling. In: ACM. **Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems**. [S.l.], 2018. p. 205.

BAI, X.; LATECKI, L. J. Path similarity skeleton graph matching. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, IEEE, v. 30, n. 7, p. 1282–1292, 2008.

CALLIGRAPHR. 2017. <https://www.calligraphr.com/>. Accessed: 2018-04-14.

CHAMBERS, E.; EMERSON, T.; GRIMM, C.; LEONARD, K. Exploring 2d shape complexity. In: **Research in Shape Analysis**. [S.l.]: Springer, 2018. p. 61–83.

CHANG, W.-D.; SHIN, J. A statistical handwriting model for style-preserving and variable character synthesis. **International Journal on Document Analysis and Recognition**, v. 15, n. 1, p. 1–19, 2012.

CHOI, H.; CHOI, S. J.; KIM, J. H. Writer dependent online handwriting generation with bayesian network. In: IEEE. **Frontiers in Handwriting Recognition, 2004. IWFHR-9 2004. Ninth International Workshop on**. [S.l.], 2004. p. 130–135.

DJEDDI, C.; GATTAL, A.; SOUICI-MESLATI, L.; SIDDIQI, I.; CHIBANI, Y.; ABED, H. E. Lamis-mshd: a multi-script offline handwriting database. In: IEEE. **2014 14th International Conference on Frontiers in Handwriting Recognition**. [S.l.], 2014. p. 93–97.

ELARIAN, Y.; ABDEL-AAL, R.; AHMAD, I.; PARVEZ, M. T.; ZIDOURI, A. Handwriting synthesis: classifications and techniques. **International Journal on Document Analysis and Recognition (IJDAR)**, Springer, v. 17, n. 4, p. 455–469, 2014.

ELARIAN, Y.; AHMAD, I.; AWAIDA, S.; AL-KHATIB, W. G.; ZIDOURI, A. An arabic handwriting synthesis system. **Pattern Recognition**, Elsevier, v. 48, n. 3, p. 849–861, 2015.

FABBRI, R.; COSTA, L. D. F.; TORELLI, J. C.; BRUNO, O. M. 2d euclidean distance transform algorithms: a comparative survey. **ACM Computing Surveys (CSUR)**, ACM, v. 40, n. 1, p. 2, 2008.

FOGGIA, P.; PERCANNELLA, G.; VENTO, M. Graph matching and learning in pattern recognition in the last 10 years. **International Journal of Pattern Recognition and Artificial Intelligence**, World Scientific, v. 28, n. 01, p. 1450001, 2014.

FONTSPACE. 2018. <http://www.fontspace.com/category/handwriting>. Accessed: 2018-04-14.

FUJIOKA, H.; KANO, H.; NAKATA, H.; SHINODA, H. Constructing and reconstructing characters, words, and sentences by synthesizing writing motions. **IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans**, IEEE, v. 36, n. 4, p. 661–670, 2006.

GIBBONS, J. D.; CHAKRABORTI, S. **Nonparametric statistical inference**. [S.l.]: Springer, 2011.

Graves, A. Generating Sequences With Recurrent Neural Networks. **arXiv e-prints**, p. arXiv:1308.0850, Aug 2013.

GUYON, I. Handwriting synthesis from handwritten glyphs. In: CITESEER. **Proceedings of the Fifth International Workshop on Frontiers of Handwriting Recognition**. [S.l.], 1996. p. 140–153.

GUYON, I.; SCHOMAKER, L.; PLAMONDON, R.; LIBERMAN, M.; JANET, S. Unipen project of on-line data exchange and recognizer benchmarks. In: IEEE. **Proceedings of the 12th IAPR International Conference on Pattern Recognition, Vol. 3-Conference C: Signal Processing (Cat. No. 94CH3440-5)**. [S.l.], 1994. v. 2, p. 29–33.

HAINES, T. S.; AODHA, O. M.; BROSTOW, G. J. My text in your handwriting. **ACM Transactions on Graphics (TOG)**, ACM, v. 35, n. 3, p. 26, 2016.

HELMERS, M.; BUNKE, H. Generation and use of synthetic training data in cursive handwriting recognition. **Pattern Recognition and Image Analysis**, Springer, p. 336–345, 2003.

HER. [S.l.]: Warner Bros. Pictures., 2013.

HOLLANDER, M.; WOLFE, D. A.; CHICKEN, E. **Nonparametric statistical methods**. [S.l.]: John Wiley & Sons, 2013.

HULL, J. J. A database for handwritten text recognition research. **IEEE Transactions on pattern analysis and machine intelligence**, IEEE, v. 16, n. 5, p. 550–554, 1994.

ISKN. 2018. <https://www.iskn.co/>. Accessed: 2018-04-16.

JAWAHAR, C.; BALASUBRAMANIAN, A. Synthesis of online handwriting in indian languages. In: SUVISOFT. **Tenth International Workshop on Frontiers in Handwriting Recognition**. [S.l.], 2006.

JIN, L.; GAO, Y.; LIU, G.; LI, Y.; DING, K. Scut-couch2009—a comprehensive online unconstrained chinese handwriting database and benchmark evaluation. **International Journal on Document Analysis and Recognition (IJDAR)**, Springer, v. 14, n. 1, p. 53–64, 2011.

JOHANSSON, S.; ATWELL, E.; GARSIDE, R.; LEECH, G. N. **The Tagged LOB corpus: users' manual**. [S.l.]: Norwegian Computing Centre for the Humanities, 1986.

JONES, M. N.; MEWHORT, D. J. Case-sensitive letter and bigram frequency counts from large-scale english corpora. **Behavior research methods, instruments, & computers**, Springer, v. 36, n. 3, p. 388–396, 2004.

LAM, L.; LEE, S.-W.; SUEN, C. Y. Thinning methodologies-a comprehensive survey. **IEEE Transactions on pattern analysis and machine intelligence**, IEEE Computer Society, v. 14, n. 9, p. 869–885, 1992.

LEE, S.-W. Off-line recognition of totally unconstrained handwritten numerals using multilayer cluster neural network. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, IEEE, v. 18, n. 6, p. 648–652, 1996.

LEONARD, K.; MORIN, G.; HAHMANN, S.; CARLIER, A. A 2d shape structure for decomposition and part similarity. In: IEEE. **2016 23rd International Conference on Pattern Recognition (ICPR)**. [S.l.], 2016. p. 3216–3221.

LIN, Z.; WAN, L. Style-preserving english handwriting synthesis. **Pattern Recognition**, Elsevier, v. 40, n. 7, p. 2097–2109, 2007.

LIU, P.; WU, Y.-J.; MA, L.; SOONG, F. K.-P. **Hidden markov model based handwriting/calligraphy generation**. [S.l.]: Google Patents, 2011. US Patent 7,983,478.

LIWICKI, M.; BUNKE, H. Iam-ondb-an on-line english sentence database acquired from handwritten text on a whiteboard. In: IEEE. **Eighth International Conference on Document Analysis and Recognition (ICDAR'05)**. [S.l.], 2005. p. 956–961.

MARTI, U.-V.; BUNKE, H. The iam-database: an english sentence database for offline handwriting recognition. **International Journal on Document Analysis and Recognition**, Springer, v. 5, n. 1, p. 39–46, 2002.

MIYAO, H.; MARUYAMA, M.; NAKANO, Y.; HANANOI, T. Off-line handwritten character recognition by svm based on the virtual examples synthesized from on-line characters. In: IEEE. **Document Analysis and Recognition, 2005. Proceedings. Eighth International Conference on**. [S.l.], 2005. p. 494–498.

MYSCRIPTFONT. 2017. <http://www.myscriptfont.com/>. Accessed: 2018-04-14.

NAKAMURA, K.; MIYAZAKI, E.; NITTA, N.; BABAGUCHI, N. Generating handwritten character clones from an incomplete seed character set using collaborative filtering. In: IEEE. **2018 16th International Conference on Frontiers in Handwriting Recognition (ICFHR)**. [S.l.], 2018. p. 68–73.

NJAH, S.; NOUMA, B. B.; BEZINE, H.; ALIMI, A. M. Mayastroun: A multilanguage handwriting database. In: IEEE. **2012 International Conference on Frontiers in Handwriting Recognition**. [S.l.], 2012. p. 308–312.

RAO, P. Shape vectors: an efficient parametric representation for the synthesis and recognition of hand script characters. **Sadhana**, Springer, v. 18, n. 1, p. 1–15, 1993.

SAABNI, R.; EL-SANA, J. Efficient generation of comprehensive database for online arabic script recognition. In: IEEE. **Document Analysis and Recognition, 2009. ICDAR'09. 10th International Conference on**. [S.l.], 2009. p. 1231–1235.

SAABNI, R. M.; EL-SANA, J. A. Comprehensive synthetic arabic database for on/off-line script recognition research. **International Journal on Document Analysis and Recognition (IJDAR)**, Springer, v. 16, n. 3, p. 285–294, 2013.

SAHA, P. K.; BORGEFORS, G.; BAJA, G. S. di. A survey on skeletonization algorithms and their applications. **Pattern Recognition Letters**, Elsevier, v. 76, p. 3–12, 2016.

SRIHARI, S. N.; CHA, S.-H.; ARORA, H.; LEE, S. Individuality of handwriting. **Journal of forensic science**, ASTM International, v. 47, n. 4, p. 1–17, 2002.

STEGMANN, M. B.; GOMEZ, D. D. A brief introduction to statistical shape analysis. **Informatics and mathematical modelling, Technical University of Denmark, DTU**, v. 15, n. 11, 2002.

STETTINER, O.; CHAZAN, D. A statistical parametric model for recognition and synthesis of handwriting. In: IEEE. **Pattern Recognition, 1994. Vol. 2-Conference B: Computer Vision & Image Processing., Proceedings of the 12th IAPR International. Conference on**. [S.l.], 1994. v. 2, p. 34–38.

THE Indian Handwritten Letter Co. 2017. <http://www.tihlc.com>. Accessed: 2018-04-14.

THOMAS, A. O.; RUSU, A.; GOVINDARAJU, V. Synthetic handwritten captchas. **Pattern Recognition**, Elsevier, v. 42, n. 12, p. 3365–3373, 2009.

TORSELLO, A.; HANCOCK, E. R. A skeletal measure of 2d shape similarity. **Computer Vision and Image Understanding**, Elsevier, v. 95, n. 1, p. 1–29, 2004.

VELTKAMP, R. C. Shape matching: Similarity measures and algorithms. In: IEEE. **Proceedings International Conference on Shape Modeling and Applications**. [S.l.], 2001. p. 188–197.

VIARD-GAUDIN, C.; LALLICAN, P. M.; KNERR, S.; BINTER, P. The ireste on/off (ironoff) dual handwriting database. In: IEEE. **Proceedings of the Fifth International Conference on Document Analysis and Recognition. ICDAR'99 (Cat. No. PR00318)**. [S.l.], 1999. p. 455–458.

WACOM. 2018. <https://www.wacom.com/>. Accessed: 2018-04-16.

WANG, J.; WU, C.; XU, Y.-Q.; SHUM, H.-Y.; JI, L. Learning-based cursive handwriting synthesis. In: IEEE. **Frontiers in Handwriting Recognition, 2002. Proceedings. Eighth International Workshop on**. [S.l.], 2002. p. 157–162.

WANG, J.; WU, C.; XU, Y.-Q.; SHUM, H.-Y. Combining shape and physical models for online cursive handwriting synthesis. **International Journal of Document Analysis and Recognition (IJDAR)**, Springer, v. 7, n. 4, p. 219–227, 2005.

WHATTHEFONT. 2017. <https://www.myfonts.com/WhatTheFont/>. Accessed: 2018-04-14.

WILCOXON, F. **Individual comparisons by ranking methods. Biom Bull 1: 80–83**. 1945.

YOURFONTS. 2017. <http://www.yourfonts.com/>. Accessed: 2018-04-14.

ZHANG, T.; SUEN, C. Y. A fast parallel algorithm for thinning digital patterns. **Communications of the ACM**, ACM, v. 27, n. 3, p. 236–239, 1984.

ZHENG, Y.; DOERMANN, D. Handwriting matching and its application to handwriting synthesis. In: IEEE. **Document Analysis and Recognition, 2005. Proceedings. Eighth International Conference on**. [S.l.], 2005. p. 861–865.

APPENDIX

APPENDIX A - HANDWRITING ACQUISITION EXPERIMENT TEMPLATE

Handwriting Experiment [1]

The following experiment consists in handwriting specific sentences using a tablet-like device called The Slate. Please bear in mind the following observations:

- You should write with the material provided by the instructor and with the paper adjusted in landscape;
- If any words were miswritten use a strikethrough (~~horizontal line through their center~~) and keep on your writing normally;
- Avoid using the dash (- symbol) to separate words between lines, except if it is specifically written in the text;
- The time you spend to complete the task is not relevant;
- When each paper is filled ask the instructor to replace the paper for a new one.

Please write on The Slate the following excerpts within the boxes using your common handwriting. Pay attention to replicate the lower and upper case letters, as well as any punctuation marks:

1. The sentence "*A quick brown fox jumps over the lazy dog.*" with only the first letter in upper case three times:

A quick brown fox jumps over the lazy dog. A quick brown fox, jumps over the lazy dog! A quick brown fox - jumps over the lazy dog?

2. The sentence "*A quick brown fox jumps over the lazy dog.*" varying the upper letters three times, the first one all in upper case, the second one as presented below:

A QUICK BROWN FOX JUMPS OVER THE LAZY DOG; "a quick brown fox jumps over the lazy dog" (A Quick Brown Fox Jumps Over The Lazy Dog)

3. The digits 0 to 9 five times:

0 1 2 3 4 5 6 7 8 9	0 1 2 3 4 5 6 7 8 9
0 1 2 3 4 5 6 7 8 9	0 1 2 3 4 5 6 7 8 9
0 1 2 3 4 5 6 7 8 9	

4. The sentences presented below with only the first letter in upper case, except the “June” word in the fourth sentence:

Pack my red box with five dozen % quality jugs The five ‘boxing’ wizards jump quickly. [Few black taxis drive up major roads on quiet hazy nights] {Back in June we delivered oxygen equipment of the same size} My girl: move six dozen plaid jackets before she quit

5. The text below:

The five boxing wizards jump quickly. Section 4. Information about Donations to the Project Gutenberg Literary Archive Foundation Editions. Aldo, Venice, 1546; della Tertina, 1550; Cambiagi, Florence, 6 vols., 1782-5; dei Classici, Milan, 10 1813; Silvestri, 9 vols., 1820-2; Passerini, Fanfani, Milanesi, 6 vols. only published, 1873-7. And without holding out his hand he walked away. "Immediately." Pprrpffrrppffff. "It's the 'cademy!" he was yelling, "the 'cademy's on fire!"

"Encouraging us to get along quicker," said another uneasily.

March. Enter EDWARD, GEORGE, RICHARD, WARWICK, NORFOLK, MONTAGUE, and soldiers

Betsy broke into unexpectedly mirthful and irrepressible laughter, a thing which rarely happened with her.

THE BEATITUDES: (Incoherently) Beer beef battledog buybull businum barnum buggerum bishop.

More quaint, more pleasing, nor more commendable; Belike you mean to make a puppet of me.

From a good distribution of enjoyments results individual happiness.

News, news from heaven! Marcus, the post is come. Sirrah, what tidings? Have you any letters? Shall I have justice? What says Jupiter?

Total, ...£1,455,949: 18: 9

And this soft courage makes your followers faint. You promis'd knighthood to our forward son: Unsheathe your sword and dub him presently. Edward, kneel down.

'important--unimportant--unimportant--important--' as if he were trying which word sounded best.

"I had not thought of that," said Alexey Alexandrovitch, evidently agreeing.

"Zarnos. Pagaiksztis. Szluofa!" (Imitative motions.)

Poulaphouca Poulaphouca Phoucaphouca Phoucaphouca.

SHALLOW. Break their talk, Mistress Quickly; my kinsman shall speak for himself.

Flourish. Enter KING HENRY, CLARENCE, WARWICK, SOMERSET, young HENRY, EARL OF RICHMOND, OXFORD, MONTAGUE, LIEUTENANT OF THE TOWER, and attendants

Thank you!

APPENDIX B - HANDWRITING ACQUISITION EXPERIMENT QUESTIONNAIRES

Agreement

You are invited to participate in an experiment for handwriting analysis using a specific hardware. Please read this document carefully and clarify your questions before agreeing to participate.

The objective of the experiment is to collect the handwriting data through a specific tablet-like device called The Slate. The collected data is anonymous and may be publicly available for research purposes. The experiment total time is around 45 minutes. You may pause for rest or quit the experiment at any time.

* Required

1. **Full name:** *

2. **E-mail** *

3. **If you agree with this term, please check the box below.** *

   Check all that apply.

   ☐ I accept to participate in this experiment.

Pre-questionnaire

4. **ID number** *

5. **Condition** *

   Mark only one oval.

   ◯ 1

   ◯ 2

6. **Age** *

7. **Education** *
Mark only one oval.

- ⬭ No schooling complete
- ⬭ High school graduate
- ⬭ Trade/technical/vocational training
- ⬭ Bachelor's degree
- ⬭ Master's degree
- ⬭ Professional degree
- ⬭ Doctorate degree

8. **Genre** *
Mark only one oval.

- ⬭ Female
- ⬭ Male
- ⬭ Other: ____________________

9. **How often do you write by hand? (minutes per week)** *

____

10. **Please mark the box that best describes which hand you use for the activity in question:** *
Mark only one oval per row.

	Always Left	Usually Left	No Preference	Usually Right	Always Right
Writing	⬭	⬭	⬭	⬭	⬭
Throwing	⬭	⬭	⬭	⬭	⬭
Scissors	⬭	⬭	⬭	⬭	⬭
Toothbrush	⬭	⬭	⬭	⬭	⬭
Knife (without fork)	⬭	⬭	⬭	⬭	⬭
Spoon	⬭	⬭	⬭	⬭	⬭
Match (when striking)	⬭	⬭	⬭	⬭	⬭
Computer mouse	⬭	⬭	⬭	⬭	⬭

11. **Are you an English native speaker?** *
Mark only one oval.

- ⬭ Yes
- ⬭ No

12. **What is your English reading skill level?** *
Mark only one oval.

	1	2	3	4	5	
Very low	⬭	⬭	⬭	⬭	⬭	Very high

13. **What is your English writing skill level?** *
Mark only one oval.

	1	2	3	4	5	
Very low	⬭	⬭	⬭	⬭	⬭	Very high

14. **What is your English speaking skill level?** *
Mark only one oval.

	1	2	3	4	5	
Very low	⬭	⬭	⬭	⬭	⬭	Very high

Post-questionnaire

15. **How natural is the writing on The Slate compared to common pen and paper?** *
Mark only one oval.

	1	2	3	4	5	
Not natural at all	⬭	⬭	⬭	⬭	⬭	Very natural

16. **How experienced are you on using tablet devices?** *
Mark only one oval.

	1	2	3	4	5	
Not experienced at all	⬭	⬭	⬭	⬭	⬭	Very experienced

17. **How natural is the writing on The Slate compared to tablet devices?**
Please answer this question only if you have ever used tablet-like devices before.
Mark only one oval.

	1	2	3	4	5	
Not natural at all	⬭	⬭	⬭	⬭	⬭	Very natural

18. **Would you like to leave some comment regarding the experiment?**

APPENDIX C - HANDWRITING ACQUISITION SUBJECTIVE EVALUATION

Análise de escrita à mão

* Required

Termo de compromisso

Você está sendo convidado a participar de um experimento sobre análise de escrita à mão. Este estudo faz parte do trabalho de pesquisa do estudante de doutorado Dennis Giovani Balreira sob orientação do professor Marcelo Walter do Instituto de Informática da UFRGS.

O objetivo do experimento é avaliar similaridades visuais de textos escritos à mão. Os dados coletados serão disponibilizados publicamente para fins de pesquisa de forma anônima. O tempo estimado do experimento é de 10 minutos. É permitido cancelar a participação durante qualquer momento durante a realização do experimento.

1. **Se você concorda com este termo, por favor marque a opção abaixo.** *

   Check all that apply.

   ☐ Eu aceito participar deste experimento.

Pré-questionário

2. **Idade** *

   ____

3. **Nível educacional** *

   Mark only one oval.

   ○ Ensino fundamental incompleto
   ○ Ensino fundamental completo
   ○ Ensino médio completo
   ○ Ensino técnico completo
   ○ Graduação completa
   ○ Especialização completa
   ○ Mestrado completo
   ○ Doutorado completo

4. **Gênero** *

   Mark only one oval.

   ○ Feminino
   ○ Masculino
   ○ Prefiro não informar
   ○ Other: ______________________

5. **Qual dispositivo você está usando para responder ao questionário?** *

Mark only one oval.

- Computador, notebook, etc.
- Smartphone, tablet, etc.
- Other: ____________________

6. **Você possui algum problema visual? Ex.: miopia, astigmatismo, hipermetropia, daltonismo, etc.** *

Mark only one oval.

- Sim
- Não

7. **Se a resposta anterior foi sim, forneça o nome do problema visual.**

____

8. **Você já participou de algum experimento relacionado com síntese de escrita à mão dos mesmos autores?** *

Mark only one oval.

- Sim
- Não

9. **Se a resposta anterior foi sim, descreva brevemente o experimento.**

____
____
____
____
____

10. **Sua língua nativa é o inglês?** *

Mark only one oval.

- Sim
- Não

11. **Qual seu nível de leitura em inglês?** *

Mark only one oval.

	1	2	3	4	5	
Muito baixo						Muito alto

12. **Qual seu nível de escrita em inglês?** *

Mark only one oval.

	1	2	3	4	5	
Muito baixo						Muito alto

13. **Qual seu nível de fala em inglês?** *

Mark only one oval.

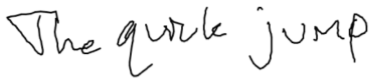
	1	2	3	4	5	
Muito baixo	( )	( )	( )	( )	( )	Muito alto

Instruções gerais

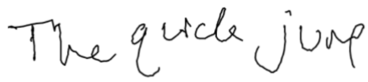
Nas questões a seguir são apresentadas duas frases de referência escritas à mão, renderizadas pelo computador. Cada frase está associada a um conjunto, denotado por A e B. Na sequência, um novo fragmento é fornecido. Sua tarefa é determinar se este fragmento pertence ao conjunto A ou ao conjunto B. Todos os trechos utilizados estão em língua inglesa.

Veja o exemplo a seguir:

Frases associadas a cada conjunto A e B:



A



B

14. **Fragmento fornecido:** *

Mark only one oval.

( ) A

( ) B

Siga para a próxima página para efetivamente dar início às questões.

Questão 1

The five boxing wizards A

The five boxing wizards B

15. **De qual conjunto o trecho abaixo foi extraído?** *

A quick brown fox

Mark only one oval.

- A
- B

Questão 2

The five boxing wizards jump quickly A

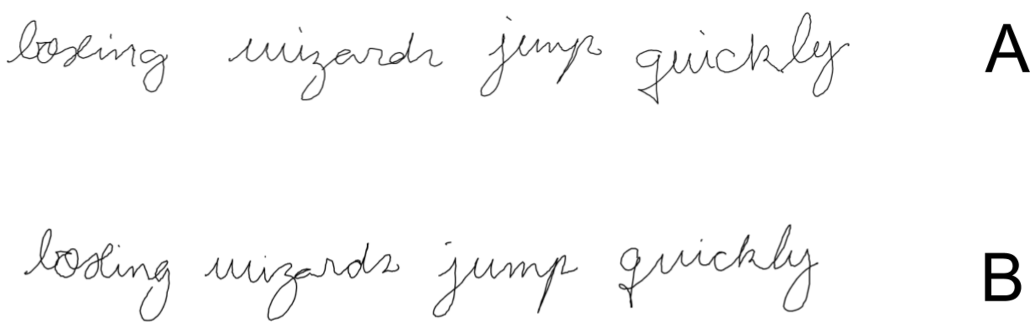
The five boxing wizards jump quickly B

16. **De qual conjunto o trecho abaixo foi extraído?** *

Mark only one oval.

- ⬭ A
- ⬭ B

Questão 3



17. **De qual conjunto o trecho abaixo foi extraído?** *

Mark only one oval.

- ⬭ A
- ⬭ B

Questão 4

The five boxing wizards jump quickly A

The five boxing wizards jump quickly B

18. **De qual conjunto o trecho abaixo foi extraído?** *

wizards jump quickly

Mark only one oval.

- [ ] A
- [ ] B

Questão 5

The five boxing wizards jump quickly A

The five boxing wizards jump quickly B

19. **De qual conjunto o trecho abaixo foi extraído?** *

happened with her

Mark only one oval.

- ( ) A
- ( ) B

Questão 6

The five boxing wizards jump quickly A

The five boxing wizards jump quickly B

20. **De qual conjunto o trecho abaixo foi extraído?** *

lazy dog

Mark only one oval.

- ( ) A
- ( ) B

Questão 7

The five boxing wizards jump quickly A

The five boxing wizards jump quickly B

21. **De qual conjunto o trecho abaixo foi extraído?** *

five dozen

Mark only one oval.

- A
- B

Questão 8

The five boxing wizards jump quickly A

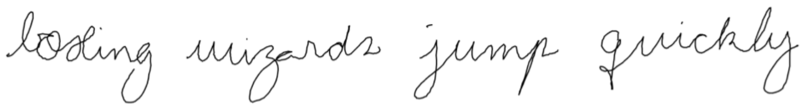
The five boxing wizards jump quickly B

22. **De qual conjunto o trecho abaixo foi extraído?** *

Mark only one oval.

- A
- B

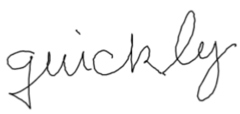
Questão 9



A

B

23. **De qual conjunto o trecho abaixo foi extraído?** *



Mark only one oval.

- A
- B

Questão 10

The five boxing wizards jump quickly A

The five boxing wizards jump quickly B

24. **De qual conjunto o trecho abaixo foi extraído?** *

courage makes your followers

Mark only one oval.

- A
- B

Questão 11

The five boxing wizards jump quickly A

The five boxing wizards jump quickly B

25. **De qual conjunto o trecho abaixo foi extraído?** *

Pack my red box

Mark only one oval.

- ( ) A
- ( ) B

Questão 12

The five boxing wizards — A

The five boxing wizards — B

26. **De qual conjunto o trecho abaixo foi extraído?** *

make a puppet

Mark only one oval.

- ( ) A
- ( ) B

Questão 13

The five boxing wizards jump quickly A

The five boxing wizards jump quickly B

27. **De qual conjunto o trecho abaixo foi extraído?** *

The five

Mark only one oval.

- A
- B

Questão 14

boxing wizards jump quickly A

boxing wizards jump quickly B

28. **De qual conjunto o trecho abaixo foi extraído?** *

holding

Mark only one oval.

- ( ) A
- ( ) B

Questão 15

The five boxing wizards jump quickly A

The five boxing wizards jump quickly B

29. **De qual conjunto o trecho abaixo foi extraído?** *

I had not thought

Mark only one oval.

- ( ) A
- ( ) B

Questão 16

The five boxing wizards jumps quickly A

The five boxing wizards jump quickly B

30. **De qual conjunto o trecho abaixo foi extraído?** *

before she quit

Mark only one oval.

- A
- B

Questão 17

The five boxing wizards jumps quickly A

The five boxing wizards jump quickly B

31. **De qual conjunto o trecho abaixo foi extraído?** *

delivered

Mark only one oval.

- ( ) A
- ( ) B

Questão 18

The five boxing wizards jump quickly **A**

The five boxing wizards jump quickly **B**

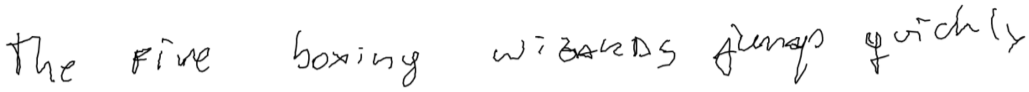
32. **De qual conjunto o trecho abaixo foi extraído?** *

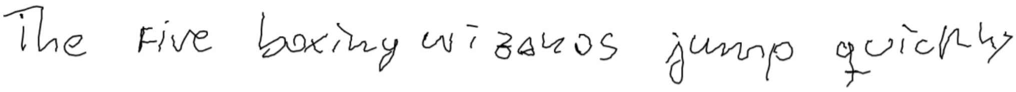
Break their talk

Mark only one oval.

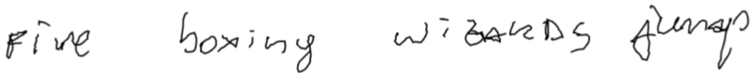
- ( ) A
- ( ) B

Questão 19

 A

 B

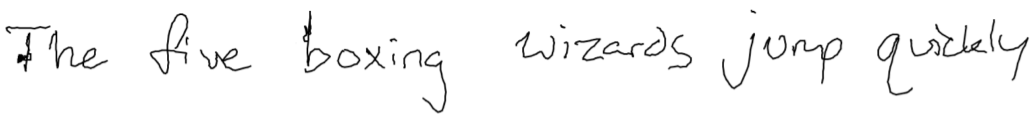
33. **De qual conjunto o trecho abaixo foi extraído?** *



Mark only one oval.

- ( ) A
- ( ) B

Questão 20

 A

 B

34. **De qual conjunto o trecho abaixo foi extraído?** *

walked away

Mark only one oval.

- ( ) A
- ( ) B

Pós questionário

35. **Na sua opinião, quão difícil foi a tarefa?** *

Mark only one oval.

	1	2	3	4	5	
Muito fácil	( )	( )	( )	( )	( )	Muito difícil

36. **Qual(is) da(s) estratégia(s) abaixo você usou para diferenciar os conjuntos?** *

Check all that apply.

- [ ] Forma dos caracteres/palavras
- [ ] Espaçamento entre caracteres/palavras
- [ ] Emparelhamento dos caracteres/palavras
- [ ] Other: ____________________

37. **Fique a vontade para colocar comentários sobre o experimento.**

____
____
____
____
____

APPENDIX D - HANDWRITING SYNTHESIS SUBJECTIVE EVALUATION

Experimento: síntese de escrita à mão com resposta

1. Coleta de dados

Idade: ____

Gênero:

( ) Masculino

( ) Feminino

Escolaridade:

( ) Fundamental - Incompleto

( ) Fundamental - Completo

( ) Médio - Incompleto

( ) Médio - Completo

( ) Superior - Incompleto

( ) Superior - Completo

( ) Outro

( ) Pós-graduação (Lato sensu) - Incompleto

( ) Pós-graduação (Lato sensu) - Completo

( ) Pós-graduação (Stricto sensu, nível mestrado) - Incompleto

( ) Pós-graduação (Stricto sensu, nível mestrado) - Completo

( ) Pós-graduação (Stricto sensu, nível doutor) - Incompleto

( ) Pós-graduação (Stricto sensu, nível doutor) - Completo

2. Sobre a tarefa

O objetivo deste experimento é comparar o nível de similaridade visual de frases formadas por diferentes traçados. Em cada etapa, uma frase principal escrita à mão se encontrará na parte destacada por um retângulo seguida por cinco frases auxiliares.

Para cada frase auxiliar selecione um grau de semelhança de 0 a 10. Quanto maior o grau, maior a semelhança com a frase principal. Os graus podem ser repetidos.

Exemplo:

FRASE PRINCIPAL

FRASE AUXILIAR 1.

0	1	2	3	4	5	6	7	8	9	10

Menor grau de semelhança — Maior grau de semelhança

FRASE AUXILIAR 2.

0	1	2	3	4	5	6	7	8	9	10

Menor grau de semelhança — Maior grau de semelhança

FRASE AUXILIAR 3.

0	1	2	3	4	5	6	7	8	9	10

Menor grau de semelhança — Maior grau de semelhança

FRASE AUXILIAR 4.

0	1	2	3	4	5	6	7	8	9	10

Menor grau de semelhança — Maior grau de semelhança

FRASE AUXILIAR 5.

0	1	2	3	4	5	6	7	8	9	10

Menor grau de semelhança — Maior grau de semelhança

3. Tarefa

Few black taxis drive up major roads on quiet hazy nights

S02-0. Frase principal.

Few black taxis drive up major roads on quiet hazy nights

0	1	2	3	4	5	6	7	8	9	10

Menor grau de semelhança — Maior grau de semelhança

s02-1. Frase auxiliar 1.

few black taxis drive up major roads on quiet hazy nights

0	1	2	3	4	5	6	7	8	9	10

Menor grau de semelhança — Maior grau de semelhança

s02-2. Frase auxiliar 2.

Few black taxis drive up Major roads on quiet hazy nights

0	1	2	3	4	5	6	7	8	9	10

Menor grau de semelhança — Maior grau de semelhança

s02-3. Frase auxiliar 3.

Few black taxis drive up major roads on quiet hAzy nights

0	1	2	3	4	5	6	7	8	9	10

Menor grau de semelhança — Maior grau de semelhança

s02-4. Frase auxiliar 4.

Few BlACK taxis drive up major roads on quiet hazy nights

0	1	2	3	4	5	6	7	8	9	10

Menor grau de semelhança — Maior grau de semelhança

s02-5. Frase auxiliar 5.

The five boxing wizards jump quickly

S03-0. Frase principal.

The five boxing wizards jump quickly

0	1	2	3	4	5	6	7	8	9	10

Menor grau de semelhança **Maior grau de semelhança**

s03-1. Frase auxiliar 1.

The five boxing wizards jump quickly

0	1	2	3	4	5	6	7	8	9	10

Menor grau de semelhança **Maior grau de semelhança**

s03-2. Frase auxiliar 2.

The five boxing wizards jump quickly

0	1	2	3	4	5	6	7	8	9	10

Menor grau de semelhança **Maior grau de semelhança**

s03-3. Frase auxiliar 3.

The five boxing Wizards jump quickly

0	1	2	3	4	5	6	7	8	9	10

Menor grau de semelhança **Maior grau de semelhança**

s03-4. Frase auxiliar 4.

The five boxing Wizards jump Quickly

0	1	2	3	4	5	6	7	8	9	10

Menor grau de semelhança **Maior grau de semelhança**

s03-5. Frase auxiliar 5.

Pack my red box with five dozen quality jugs

S04-0. Frase principal.

Pack my red box with five dozen quality jugs

0	1	2	3	4	5	6	7	8	9	10

s04-1. Frase auxiliar 1. Menor grau de semelhança Maior grau de semelhança

Pack my red box with five dozen quality jugs

0	1	2	3	4	5	6	7	8	9	10

s04-2. Frase auxiliar 2. Menor grau de semelhança Maior grau de semelhança

Pack my red box with five dozen quality jugs

0	1	2	3	4	5	6	7	8	9	10

s04-3. Frase auxiliar 3. Menor grau de semelhança Maior grau de semelhança

Pack my red box with five dozen quality jugs

0	1	2	3	4	5	6	7	8	9	10

s04-4. Frase auxiliar 4. Menor grau de semelhança Maior grau de semelhança

Pack my red box with five dozen quality jugs

0	1	2	3	4	5	6	7	8	9	10

s04-5. Frase auxiliar 5. Menor grau de semelhança Maior grau de semelhança

Back in June we delivered oxygen equiptment of the same size

S06-0. Frase principal.

Back in June we delivered oxygen equiptment of the same size

0	1	2	3	4	5	6	7	8	9	10

Menor grau de semelhança — Maior grau de semelhança

s06-1. Frase auxiliar 1.

Back in June we delivered oxygen equiptment of the same size

0	1	2	3	4	5	6	7	8	9	10

Menor grau de semelhança — Maior grau de semelhança

s06-2. Frase auxiliar 2.

Back in June we delivered oxygen equiptment of the same size

0	1	2	3	4	5	6	7	8	9	10

Menor grau de semelhança — Maior grau de semelhança

s06-3. Frase auxiliar 3.

Back in June we delivered oxygen equiptment of the same size

0	1	2	3	4	5	6	7	8	9	10

Menor grau de semelhança — Maior grau de semelhança

s06-4. Frase auxiliar 4.

Back in June we delivered oxygen equiptment of the same size

0	1	2	3	4	5	6	7	8	9	10

Menor grau de semelhança — Maior grau de semelhança

s06-5. Frase auxiliar 5.

Pack my box with five dozen liquor jugs

S09-0. Frase principal.

Pack my box with five dozen Liquor jugs

0	1	2	3	4	5	6	7	8	9	10

Menor grau de semelhança — Maior grau de semelhança

s09-1. Frase auxiliar 1.

Pack my box with five dozen liquor jugs

0	1	2	3	4	5	6	7	8	9	10

Menor grau de semelhança — Maior grau de semelhança

s09-2. Frase auxiliar 2.

Pack my box with five dozen liquor jugs

0	1	2	3	4	5	6	7	8	9	10

Menor grau de semelhança — Maior grau de semelhança

s09-3. Frase auxiliar 3.

Pack my box with five dozen liquor jugs

0	1	2	3	4	5	6	7	8	9	10

Menor grau de semelhança — Maior grau de semelhança

s09-4. Frase auxiliar 4.

Pack my box with five dozen liquor jugs

0	1	2	3	4	5	6	7	8	9	10

Menor grau de semelhança — Maior grau de semelhança

s09-5. Frase auxiliar 5.