QUANTIFYING SCRIBAL BEHAVIOR: A NOVEL APPROACH TO DIGITAL PALEOGRAPHY

Vinodh Rajan Sampath

A Thesis Submitted for the Degree of PhD at the University of St Andrews

2016

Quantifying Scribal Behavior: A Novel Approach to Digital Paleography

by

Vinodh Rajan Sampath

University of
St Andrews

FOUNDED
1413

This thesis is submitted to the
University of St Andrews
in partial fulfilment for the degree of
Doctor of Philosophy
submitted on

August 2016

Abstract

We propose a novel approach for analyzing scribal behavior quantitatively using information about the handwriting of characters. To implement this approach, we develop a computational framework that recovers this information and decomposes the characters into primitives (called strokes) to create a hierarchically structured representation. We then propose a number of intuitive metrics quantifying various facets of scribal behavior, which are derived from the recovered information and character structure. We further propose the use of techniques modeling the generation of handwriting to directly study the changes in writing behavior.

We then present a case study in which we use our framework and metrics to analyze the development of four major Indic scripts. We show that our framework and metrics coupled with appropriate statistical methods can provide great insight into scribal behavior by discovering specific trends and phenomena with quantitative methods. We also illustrate the use of handwriting modeling techniques in this context to study the divergence of the Brahmi script into two daughter scripts.

We conduct a user study with domain experts to evaluate our framework and salient results from the case study, and we elaborate on the results of this evaluation. Finally, we present our conclusions and discuss the limitations of our research along with future work that needs to be done.

Candidate's Declaration

I, Vinodh Rajan Sampath, hereby certify that this thesis, which is approximately 33,000 words in length, has been written by me, and that it is the record of work carried out by me, or principally by myself in collaboration with others as acknowledged, and that it has not been submitted in any previous application for a higher degree.

I was admitted as a research student in September 2012, and as a candidate for the degree of Doctor of Philosophy in May 2016; the higher study for which this is a record was carried out in the University of St Andrews between 2012 and 2016.

Date ..... Signature of Candidate .....

Supervisor's Declaration

I hereby certify that the candidate has fulfilled the conditions of the Resolution and Regulations appropriate for the degree of Doctor of Philosophy in the University of St Andrews and that the candidate is qualified to submit this thesis in application for that degree.

Date ..... Signature of Supervisor .....

Permission for publication

In submitting this thesis to the University of St Andrews I understand that I am giving permission for it to be made available for use in accordance with the regulations of the University Library for the time being in force, subject to any copyright vested in the work not being affected thereby. I also understand that the title and the abstract will be published, and that a copy of the work may be made and supplied to any bona fide

library or research worker, that my thesis will be electronically accessible for personal or research use unless exempt by award of an embargo as requested below, and that the library has the right to migrate my thesis into new electronic forms as required to ensure continued access to the thesis. I have obtained any third-party copyright permissions that may be required in order to allow such access and migration, or have requested the appropriate embargo below.

The following is an agreed request by candidate and supervisor regarding the publication of this thesis:

No embargo on any electronic nor print copy.

Date ..... Signature of Candidate .....

Signature of Supervisor .....

𑀇𑀬𑀁 𑀥𑀫𑀫𑀮𑀺𑀧𑀺 𑀤𑁂𑀯𑀦𑀁𑀧𑀺𑀬𑁂𑀡 𑀧𑀺𑀬𑀤𑀲𑀺𑀦𑀸 𑀮𑀸𑀚𑀺𑀦𑀸 𑀮𑀺𑀔𑀸𑀧𑀺𑀢𑀸

iyaṃ dhaṃmalipi devānaṃpiyena piyadasinā lājinā likhāpitā

This writing of *dharma*[1] was caused to be written by king Devānaṃpiya Piyadasi (Aśoka)

From the inscriptions of king Aśoka, around 3rd century BCE

This thesis is dedicated to those who carved these edicts and helped spread writing all over the Indian subcontinent two millennia ago

[1]This is an extremely overloaded term that is usually translated as either law or religion. In this specific context it probably carries the connotation of *righteous way/order [of life]*.

Acknowledgements

My journey towards this thesis has been long and mostly pleasant but not without the usual highs and lows that come as part and parcel of the journey itself. This document wouldn't have been made possible if not for the following people who supported me along the way. First and foremost would be my supervisor Dr Mark-Jan Nederhof. He has been an excellent guide and a wonderful mentor hugely supporting my research and providing constructive critiques when necessary. At times, he was the one who lent an ear when I ranted about my research. The entire thesis and my research reached the current shape because of his constant guidance. Given my background as an engineer used to building things, Dr Nederhof had a major part in mentoring me to be a researcher.

I also should mention a few others without whom I would not have started this PhD. Dr Jean-Luc Chevillard (CNRS, Université Paris-Diderot) was the one who initially prodded me into doing a PhD and has been a gentle guide and mentor all along. Dr Shriramana Sharma (Nerur, India), a long time friend and collaborator, supported me a lot with my PhD application. ಕನ್ನಡ ಪುಸ್ತಕಗಳಿಗೆ ಧನ್ಯವಾದಗಳು. My mate Prem Krishnan provided me with the much needed academic papers at unearthly hours while I was writing my PhD proposal.

SICSA provided the necessary funding for this research through their Prize PhD Studentship. I am grateful to my second supervisor Dr Aaron Quigley, my previous second supervisor Dr Per Ola Kristensson (University of Cambridge), and my reviewers Dr Graham Kirby, Dr Chris Jeffer-

son and Dr John Thomson for their feedback. I am also grateful to all the peer-reviewers of my publications whose comments helped to improve and focus my research. The school administration team, especially Ms Judi Robertson, Ms Julie Dunsire and Mr Alex Bain, helped with all the day-to-day needs and formalities of my research and conference travel.

Dr Réjean Plamondon (École Polytechnique de Montréal) provided the software that implemented the Sigma-Lognormal model. Thanks to invitations from Dr Eva Wilden (CSMC, University of Hamburg), I was able to visit the University of Hamburg twice to present my work and exchange fruitful discussions at the Centre for the Study of Manuscript Cultures. I was also able to visit the University of Cambridge thanks to Dr Vincenzo Vergiani, who kindly invited me to discuss my research and provided valuable manuscript data.

I also utilize this space for everyone who made my journey towards my PhD a lot more interesting and enjoyable. My officemates Shyam Reyal and Martin McCaffery provided a conducive environment and the much needed distraction (and occasional support) for my research. *Merci* & ಸ್ತುತಿ to you both! I would like to mention my Scottish Gaelic teacher Mòrag, with whom I studied Gaelic for two years, for providing a refreshing break from my research. *Tapadh Leibh, a Mhòrag! Bu mhòr a chòrd na clasichean agaibh rium.*

I am indebted to my parents, my uncle and aunt, and in fact to all of my extended family, without whose support I wouldn't be here. మీ అందరికి నా కృతజ్ఞతలు. I am also thankful to my close friends in Scotland who have supported and been with me when I was going through a very tough phase in my life.

I guess I kept the last spot for the best. My boyfriend (and fellow PhD student) Juan José Mendoza Santana was with me all through the writing stage and coped up with my terrible mood swings. Without his constant support, de-stressing and motivation I would not have made through the writing phase sane. *Eres el mejor. Sólo quiero decirte que estoy muy feliz por haberte encontrado en el momento perfecto y haber entrado en mi vida. Muchas gracias por estar siempre conmigo y apoyarme, cariño mío. ¡Sin ti, estaría perdido! Sólo desearía poder pasar el resto de mi vida junto a ti in sæcula sæculorum.*

Contents

Chapter 1

Introduction

> This invention [Writing], O king... will make the Egyptians wiser and will improve their memories; for it is an elixir of memory and wisdom that I have discovered.
>
> Thoth, Egyptian god of writing to Thamos, king of Egypt[1]

1.1 Motivation

Paleography as a discipline concerning historical handwriting probably dates back to the times of the Benedictine monks of St Maur, when they started cataloging handwritten manuscripts three centuries ago (Aussems & Brink, 2009). Ever since, the discipline has firmly established itself as an integral part within the broader area of history. Paleography as a field has evolved substantially in these three hundred odd years. There were multiple attempts to move it from the realm of subjectivity[2] to objectivity[3], as progressive efforts were made to

[1] From *Plato's Phaedrus* (Fowler, 1925)

[2] This denotes here overt dependence on personal internal opinions and intuitions that cannot be explicitly stated and substantiated through external facts and evidences.

[3] This denotes here dependence on external facts and evidences. True objectivity is hard to achieve, as something is, more often than not, always interpreted by a human at some level. See Daston and Galison (2007) for a detailed view on objectivity and (Sculley & Pasanek, 2008) in relation to digital humanities.

introduce methodological approaches to paleographic analyses starting from the middle of the twentieth century such as Mallon (1952) and Gilissen (1973). The term *Digital Paleography* was originally coined only in 2005 (Ciula, 2005), to describe an innovative approach that applied computational methods to paleography. The very late rise of digital methods is not surprising as the discipline was long held to be an inexact science; an art of appreciating aesthetics that defies objectification and thus quantification. Not surprisingly, quantitative methods are still seen with skepticism, and even now the trained eye of a paleographer is generally trusted more. As such, Digital Paleography does not yet have a wider adoption into the mainstream field of paleography.

On a wider scale, scripts are often seen as simple carriers of languages and usually relegated to an auxiliary role. Research on scripts until recently has been minimal and niche, except for the field of paleography. They are an important part of the cultural heritage of humanity and their analysis and study require more research. Fortunately, there is a growing interest in the analysis of scripts per se. Altmann and Fengxiang (2008) published a volume titled *Analyses of Scripts: Properties of Characters and Writing Systems* to explore various properties such as complexity, ornamentality and distinctivity. Changizi et al. (2006) discuss the various contour configurations of written symbols and their similarity to the environment in which they were produced. They also study the distribution of the configurations of various scripts. Changizi and Shimojo (2005) further discuss complexity of characters and the redundancy of stroke combinations of various major writing systems across history. It is to be noted that analyses by Changizi and most methods described in Altmann and Fengxiang (2008) are performed manually. Traditionally, analysis and study in paleography have also been done manually. As mentioned earlier, digital paleographic methods are at present making more inroads into the field. However, applying quantitative analysis on paleographic data is not yet popular or standardized (Stokes, 2009a). Paleographers rarely use such approaches and tend more towards semantic approaches. One of the reasons is that these systems are seen as *black boxes*, where both the approach and also the underlying assumptions are hard to understand and test (Stokes, 2012).

We think this is partially due to the difficulty of quantifying script related features, and partially due to the lack of defined methods and metrics with theoretical and qualitative underpinnings. Hence, there is a distinct need for metrics and methods, which can be used in quantitative analysis but still have sound semantic and qualitative interpretations. This helps the researcher to perform the analysis efficiently and at the same time have a better understanding of the methods and metrics that are being employed. Such an approach must also lend itself to the application of computational techniques, which would enable users to perform analyses with much more ease. Thus, a well grounded computational approach would be extremely helpful in encouraging analysis of scripts, within and outside the context of paleography.

Paleography is traditionally associated with the classification of letters, which usually takes the form of assigning provenance to a scribal artifact. One of the most important tasks of a paleographer is to assign a scribe, a geographical location or a time period to a piece of handwritten artifact. This is usually expressed as detection of (scribal) hands. Stansbury (2009) classifies paleography into two different approaches - the *Linnaean* approach and the *Darwinian* approach. The former is more focused on classification as we just saw and the latter focuses on explaining "the ways that scribes created and modified scripts". We could call this approach *Descriptive Paleography*. It can explain fundamental paleographic phenomena such as "the evolution of scripts and their relationships to each other" as well as "looking for mechanisms to explain these phenomena".

Jean Mallon was a pioneer in what Stansbury refers to as Darwinian paleography. Mallon et al. (1939) attempts to explain the evolution of Latin minuscule letters from the corresponding majuscule letters. While elaborating the physical processes that drive the evolution, he noted "This evolution of the majuscule to minuscule proceeds under the influence of the muscles of the hand, which always traces the strokes of the capital in the same order, then over time joins them, rounding and simplifying them under the control of the eye"[4] (Mallon, 1937). Also, Peignot (1937)

[4]cette évolution de la capitale à la minuscule se déroule sous l'influence des muscles de la main, qui trace toujours les traits de la capitale dans le même ordre, puis, petit à petit, les unit, les arrondit et les simplifie, et ce sous le contrôle de l'œil

while discussing the same evolution process observed that some lower case letters retain the form of their capital letters "only because these simple forms were easily written, and that scribe's hands did not feel the need to simplify"[5]. It can be noticed that even in the early 20th century, paleographers were interested in understanding the physical processes that produce handwriting and used it to explain the evolution of characters from their perspective.

Figure 1.1: Evolution of Latin minuscule from majuscule as derived by Mallon (pointypo, 2013)

Blanchard (1999), following Mallon, explains the evolution of lower case Greek letters in terms of changes in handwriting and stroke behavior. He formulates a concept called *The Unit Ductus*[6] that is "unchanged across the ages". He notices the various changes that occur to characters' strokes such as rounding of corners, change in angles, merging of strokes, etc. He uses them along with the reconstructed handwriting motion to explain the evolution of minuscule Greek letters from their corresponding majuscule letters.

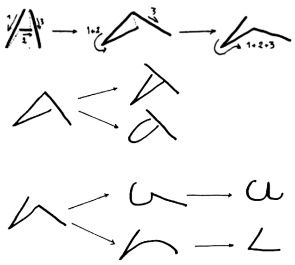


Figure 1.2: Evolution of Greek minuscule letter 'α' from majuscule 'A' (Blanchard, 1999, 8)

[5]c'est uniquement parce que ces formes simples s'écrivaient facilement et que la main des scribes n'a pas éprouvé le besoin de les simplifier.

[6]L'unité de Ductus

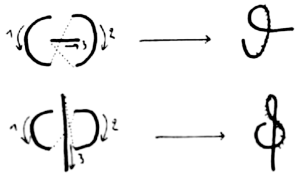


Figure 1.3: Evolution of Greek minuscule letter 'ϑ' from majuscule 'Θ' and minuscule 'φ' from majuscule 'Φ' (Blanchard, 1999, 20)

Characters in paleography (and outside of it) have for the most part been analyzed as inanimate forms. But in fact, characters inherently contain information regarding movements of the writing implement (*i.e. Ductus*), which animates them and provides profound information about their formation and therefore about scribes. Canart (2006) has also argued that common paleographic procedures only allows us to see the static aspect and we could use appropriate descriptions that can serve as clues to the dynamic aspect of writing. While focusing on the movements of implements may not appear to be outright applicable to the Linnaean approach, it is fundamental for the Darwinian approach as seen above. In fact, Stansbury (2009) also notices the lack of handwriting-motion based analysis in digital paleography and says:

> *"[...] manuscripts have the potential to deliver up a vast quantity of information about how scribes wrote. Perhaps the automated analysis of script will soon turn its attention to reconstructing the motion of the scribe's pen on the page and [...] explore the ways that these strokes evolved. It is then that we will begin to be able to measure the scribe's art."*

This thesis is an attempt to measure *the art of a scribe* by proposing methods that can quantify their writing behavior. We saw that both Mallon and Blanchard provide a basic explanation of the paleographic evolution of characters by manually analyzing them. But given the advancement in the field of computer science, it is possible to develop a comprehensive framework that would provide techniques to perform such analyses computationally. This would enable us to analyze various paleographic phenomena very effectively in a wide variety of ways and

also in much more detail. Stansbury (2009) correctly notes that the Darwinian approach may have the "most interesting collaboration with technology". In a similar manner, Aussems and Brink (2009) additionally point out that an inter-disciplinary approach could lead to ways that change "the way we look at scribal hands and medieval handwriting", moving away from paleography in its traditional form. In fact, this thesis focuses and expands on such a framework, which would provide metrics and methods that can be used to explain changes in strokes and the relationship between characters.

However, these approaches need not necessarily be seen as mutually exclusive with the Linnaean approach. This attempt to *quantify* and *assist* descriptive paleography, while helping to understand scribal behavior, can also greatly aid in the classification of scripts. By focusing on the fundamental nature of these characters, we can propose methods that are more intuitive and semantic, which the current digital paleographic methods usually lack.

1.2 Hypothesis

The central hypothesis of this thesis is that information on the handwriting of characters (either recovered or injected into paleographic data) provides insights into scribal behavior and assists in creating intuitive computational methods for quantitative digital paleographic analyses.

1.3 Research Questions

In order to validate the central hypothesis, we seek to answer the following research questions.

1. What is the computational framework required to extract handwriting information present in characters and analyze them?

2. What metrics and methods are required to perform quantitative analysis of scribal behavior using the extracted information?

3. How can the recovered information be directly used to quantitatively study changes in scribal behavior?

1.4 Contributions

We make the following contributions through this thesis.

1. We propose a modular framework that performs recovery of characters' handwriting information and decomposes the characters into proper primitives suitable for digital paleographic analysis.

2. We propose a range of intuitive quantitative metrics (and accompanying statistical methods) that can quantify handwriting information contained in characters and be used to analyze scribal behavior.

3. We propose the use of the Sigma-Lognormal model of handwriting generation to quantitatively analyze shape changes (and hence scribal behavior) in scripts.

1.5 Limitations

Scripts can generally be divided into *connected* and *unconnected* writing. In connected writing, such as cursive Roman, individual characters are conjoined, thereby reducing pen-lifts and making writing faster. This form of connected writing is commonly referred to as *cursive* style (but there are cursive styles such as Ancient Egyptian Hieratic where individual letters are not necessarily connected). The approach proposed in this thesis is ideally designed to work with writing styles that have unconnected characters. If the metrics are to be adapted for connected writing, the writing would have to be segmented into individual characters as part of preprocessing and then analyzed. Alternatively, if the analysis is performed on the individual base graphemic (hence unconnected) set instead of character exemplars from manuscripts/epigraphs, the metrics can be readily applied to any script.

The methods described in this thesis may also not be ideal for *block* scripts such as Chinese or Korean Hangul. In these systems, the base

graphemes (radicals in Chinese and jamo in Hangul) are arranged in blocks to form characters. Even though a single block can be considered as a *character* for practical purposes, specific methods based on the arrangement of graphemes might be better suited for such block scripts.

1.6 Publications

Some material presented in this thesis has also appeared as the following peer-reviewed publications.

1. Rajan, V. (2016). Quantifying Scripts: Defining metrics of characters for quantitative and descriptive analysis. Digital Scholarship in the Humanities. [To Appear]

2. Rajan, V. (2015). How Handwriting Evolves: An Initial Quantitative Analysis of the Development of Indic Scripts. Proceedings of the 17th International Graphonomics Society Conference.

3. Rajan, V. (2014). Framework for Quantitative Analysis of Scripts. DH2014 Book of Abstracts, Digital Humanities 2014.

1.7 Structure

The structure of this thesis is outlined below.

Chapter 2 discusses the background literature pertaining to the methods employed in the thesis, along with the previous work that is related to our work. It also discusses the shortcomings of the previous approaches.

Chapter 3 describes the methodology that is being proposed in this thesis. It describes a script analysis framework to analyze scripts using the recovered trajectory information of characters. It also details various metrics that are derived using the framework for quantitative and descriptive analysis. It further discusses how handwriting modeling techniques can be utilized in the context of analyzing shape change of characters.

Chapter 4 shows a case study where the proposed framework and metrics are applied to analyze the development of Indic scripts. It also

illustrates the use of handwriting modeling to understand changes in Indic scribal behavior.

Chapter 5 presents the evaluation of the framework, the metrics and some of the analyses of our case study.

Chapter 6 summarizes the thesis and presents our conclusions. It also establishes the limitations of our work and proposes future work that needs to be done.

Appendix A: University Ethics Approval for User Evaluation Study
Appendix B: Questionnaire for User Evaluation Study

Chapter 2

Background and Related Work

> You have invented an elixir not of memory, but of reminding; and you offer your pupils the appearance of wisdom, not true wisdom, for they will read many things without instruction and will therefore seem to know many things [...] since they are not wise, but only appear wise.
>
> Thamos to Thoth[1]

2.1 Digital Paleography

Digital Paleography is a fairly recent term dating back to 2005, as seen in section 1.1. Before we discuss it, we must first establish the context for quantitative methods in paleography, which by extension forms the groundwork for the application of digital methods. Some early examples of which include Loew (1914) and Mallon (1952). Loew (1914) described in detail various criteria for dating and localizing such as abbreviations and variant letter forms. This was followed by the significant work of Mallon (1952), who proposed seven factors that must be considered while distinguishing scribal hands. More than a decade later, Meuthen and Prevenier (1968) made some additions to Mallon's original list, which are generally considered redundant. However, they

[1] From *Plato's Phaedrus* (Fowler, 1925)

found that some factors suggested by Mallon such as angle of writing could also be expressed quantitatively. But it was Gilissen (1973) who prominently proposed the methodological quantification of those factors. He defined methods of analysis and an investigative process for each of those methods. He also tried to apply the scientific method to paleography by creating formulas for some of the factors rather than relying on their verbose descriptions (see §2.2 for more details). As discussed previously in section 1.1, paleography has been mostly viewed as an art that is to be imparted through subjective analysis rather than a science that can be objectified (Gumbert, 1976; Costamagna et al., 1995; Pratesi, 1998). This lack of trust in objectifying paleography is also extended to the application of quantitative methods, which by design requires the objectification of methodology. Not surprisingly, there were reservations among paleographers to accept newly proposed quantitative methods in paleography. Poulle (1974) critiqued Gilissen's methodology and was skeptic about the universality of the methods, noting several deficiencies in his criteria. However, he finally acknowledged the contribution as a turning point. A similar critique is provided in d'Haenens (1975). Around the same time, Bischoff and Koch (1979) correctly predicted that due to (advancements in) technical means paleography was on the path of becoming an art of measurement from being an art of aesthetics (Stansbury, 2009). This indeed was to become true in the future. Derolez (2003) comments that the existing methods in paleography tend to be overtly subjective, often depending upon the authority of the author and the faith of the reader (Stokes, 2009a). He proposes replacing the qualitative techniques with quantitative ones. Aussems (2006) coins the term *Scribal Fingerprint* to denote objective and quantifiable characteristics that are unique to a scribe. He also uses this to perform quantitative paleography on a medieval manuscript. He confirms that numerical techniques can indeed be very convenient by facilitating a quick but at the same time more accurate and objective analysis (Aussems & Brink, 2009).

Digital paleography is a very diverse area. Systems designed to enhance/recover images, classify/date characters, construct facsimiles, etc. can all be categorized under the umbrella of digital paleographic systems. We restrict ourselves here to the systems that focus on analyzing

characters, as this is relevant to the area of our interest in this thesis. Through these systems, we intend to showcase the growing need to have a combined quantitative and qualitative approach to analyze characters.

In terms of quantitative methods in digital paleography, A *System for Palaeographic Investigations* (SPI) (Ciula, 2005, 2009; Aiolli & Ciula, 2009) was developed at the University of Pisa to help paleographers to classify and identify scripts. In fact, the term *Digital Paleography* was coined to describe this attempt. The system aims to provide quantitative support to analyze unseen documents in the context of documents already processed by it. The system consists of several modules that perform the processing. The segmentation module allows users to extract individual characters from manuscripts. These extracts are then used by the analysis module to create, what they refer to as, tangent-based models for each character by essentially averaging them. Relationships between a new sample and data already existing in the paleographic database (through the constructed models) are also given by this module. The system additionally allows users to morph and visualize transformations of a character. Bulacu and Schomaker (2007b, 2007a) develop a writer identification tool called the *Groningen Automatic Writer Identification System* (GRAWIS) that uses probability distribution functions (PDFs) to characterize individual writers. Various PDFs were used to encode both textural and allographic features. Other similar work such as Bulacu et al. (2003) and Bulacu and Schomaker (2006) are also of interest. Stokes (2007) proposes an analysis of scribal hands through image-processing and data-mining. He extracts features based on pixel information to perform quantitative paleographic analyses. He selects five different features (extracted from forensic recognition such as Bulacu and Schomaker (2007b)) and tests their usefulness for studying medieval handwriting. The features are used for a clustering algorithm in an experiment trying to group related samples of handwriting, which is largely successful with few errors. He also suggests that tools for paleographic analysis must be used with caution but can be effective to supplement human judgments. Stokes also releases a program called *Hand Analyzer* (Stokes, 2009b, 2009a) that follows the principles outlined in Stokes (2007). The system is modular and extensible and generates a *hand* file that contains

quantitative features to describe scribal hands, based on images. The files then can then be used for measuring statistical distances between two scribal hands. Azmi et al. (2011) perform digital paleographic analysis of Jawi manuscripts for classification of writing styles through features extracted from constructing scalene triangle blocks. Similarly, Soumya and Kumar (2014) attempt to classify ancient Indian epigraphs using random forests based on their time period. Similar approaches have also been performed on Greek inscriptions in Papaodysseus et al. (2010). There are several work such as Wolf et al. (2011) that invoke complex statistical processes to attempt the classification of characters. We do not enumerate them all.

Most of these systems do not aim at extracting information that is easily understood by paleographers. They use *black box* like features, which are not readily interpretable by them. Hassner et al. (2013) point out that "high-level terminology, natural to paleography, should be integrated into computerized paleographic systems". This is partially explored by Brink et al. (2012), who propose a new feature called *Quill*. The Quill feature uses the relation between the ink direction and the ink width to identify writers. The feature aims to be intuitive and easily explainable as it is directly derived through the modeling of trace production by a quill. Herzog et al. (2010) propose a completely autonomous system to extract strokes from historical scripts using Constrained Delaunay Triangulation (CDT), motivated by stroke extraction procedures in Chinese script. The extracted strokes are meant to be used as a proxy for shape features in the context of various paleographic analyses. However, they do not propose any further methods for analyzing the information. Also, the strokes are at a very high level and are not very effective in characterizing the handwriting process of the characters (which we are more interested in).

There have been attempts that may be categorized under descriptive analysis as well. Stokes (2012) proposes a conceptual model that describes handwriting information into a hierarchical class-based system consisting of graphemes at the top level and characters, allographs, ideographs, scribal hands, etc. at subordinate levels. Each of these classes attempts to capture information (provided by the user) at its particular level of abstraction through descriptive labels (although quantitative metrics could

also be adopted). This approach mainly aims at character retrieval and search at different levels of character abstractions using string based attributes rather than proper quantitative analysis. In Levy et al. (2012), characters were manually described using a standard set of string-based descriptors. These are then analyzed for the *distinctive nature* of their appearance using Gene Set Enrichment Analysis (GESA). They show that by using appropriate statistical methods, it is possible to get meaningful insights in paleography. Stokes et al. (2014) present a script framework called *Digipal* that attempts to provide clear and convincing paleographical descriptions by using a formal model for describing handwriting. This involves tagging the manually segmented characters with structured string descriptors. Rather than performing qualitative or quantitative analysis, Digipal aims to be an exploratory and also a pedagogical tool. While all these methods lean towards descriptive analysis, they are also very qualitative and do not invoke quantitative features very much.

We can see that descriptive systems are mostly not quantitative, and quantitative systems are not necessarily descriptive. Also, the *ductus* feature, which can be defined as the direction and order of strokes to produce a character, has not been given high priority (nor has it been the basis of analysis) in most of the systems discussed. Apart from a few systems that focus on descriptors for scribal hands, most of the systems do not focus on deriving a descriptive analysis of them. This is understandable as these systems are more interested in classification. Hence, there is a distinct need for a system that enables descriptive digital paleography based on quantitative analysis. In the next section, we look into both paleographic and non-paleographic features that were proposed to analyze characters.

2.2 Character Features

As discussed in the previous section, there have been several proposals to objectify paleographic evaluation. We begin by expanding on the differentiators proposed by Mallon (1952) to perform objective paleography. He enumerated a list of seven aspects (Stokes, 2009a; Aussems & Brink, 2009)

that must be considered when trying to distinguish between various scribal hands.

1. Form (morphology of letters)
2. Angle of writing (in relation to the base line)
3. Ductus
4. Modulus (proportions of letters)
5. Contrast (the difference in thickness between the hair lines and the shadow lines)
6. Writing support
7. Internal characteristics (nature of the text)

However, they are originally meant to be descriptors rather than quantified values. Others that follow tend to be similar either defining additional criteria or requiring more details. Many of these criteria are text-based and/or linguistic differentiators such as orthography, abbreviation, punctuation, etc. As described earlier, it is Gilissen (1973) who prominently proposed quantifying features such as modulus and the angle of writing. There also have been several variants and improvements of Mallon's fundamental differentiators such as M. P. Brown (1996), Rumble (1994) and T. J. Brown (1993). Burgers et al. (1995) using the existing objective features of paleography and including some aspects from forensic analysis comes up with a methodology appropriately named as *Burger's Methodology*. The methodology is adapted (with minor modification) to medieval manuscripts in Aussems (2006), which also contains a detailed analysis of each feature in it. He chooses eight features to analyze, out of which four are quantifiable. They are as follows: (i) Angle of inclination (ii) Angle of writing (iii) Modulus (iv) Degree and type of curvisation of connecting characters (Aussems & Brink, 2009). At this point, we can comfortably invoke the term *features*, a term frequently used in machine learning to describe numerical measurements of sorts, which quantify various aspects of the object under consideration. While

we eschew the term at least in the context of paleography (in the future chapters), it is frequently used in the context of computer science. A significant overview of major paleographic features that have been proposed over time can be seen in Stokes (2009a) and Aussems (2006).

It must be noted that *ductus* is usually included in many of the feature sets but never given importance. In fact, there are no major feature sets that include ductus as the main factor, or list other major factors, save for a few, that are based or derived mainly from the ductus feature. It was generally thought to be either not quantifiable or not considered to be very usable for paleographic analyses. It is also highly difficult to recover it from an image of a specimen of writing. Apart from paleography, several other fields are also interested in quantifying characters through various features, which we elucidate below.

The most related area to digital paleography is automatic forensic document analysis. In this, a handwriting sample is usually compared to other samples to identify the writer of the sample in question. Such systems can identify, within a degree of uncertainty, if two documents were written by the same person or not(Impedovo & Pirlo, 2008). This is not very different from the detection of hands performed in paleography, and the related methodologies from forensic analysis can, in theory, be directly applied to it. Some of the digital paleographic systems described earlier were based on such methods. However, most forensic systems work as black boxes and rely on complex statistical methods that are hard to understand (T. Davis, 2007; Hassner et al., 2013). The methods are also mostly based on the static image of a sample and do not involve the handwriting information. As interesting as it may be, classification of scripts is not main motive of our research. Hence, we will not be focusing on features used in these systems, and interested readers may refer to literature reviews in related work such as Bulacu and Schomaker (2007a) that discuss applications of forensic methods to paleography. An interesting discussion comparing and contrasting paleographic and forensic handwriting identification can be seen in T. Davis (2007).

From a purely linguistic perspective, the volume published by Altmann and Fengxiang (2008) proposes various properties of characters and writing systems. Several properties such as complexity, ornamentality, dis-

tintinctivity were discussed in detail. This work is done in the context of quantitative linguistics to derive quantitative descriptions of writing systems. Interestingly, some of the metrics proposed such as distinctivity (Antić & Altmann, 2005) are very much applicable in the context of paleography. For instance, Hegenbarth-Reichardt and Altmann (2008) apply one of the metrics that define *complexity* to the study of the development of Egyptian Hieratic from Egyptian Hieroglyphs. By quantifying complexity, they analyze the process of simplification of Hieroglyphs and attempt to fit a mathematical model for the process.

Handwritten gestures are quickly becoming a very common way to interact with various devices (Mitra & Acharya, 2007). As a result, a large amount of research has been performed in terms of quantifying gestures, mostly in the context of their recognition through machine learning. Applying machine learning techniques typically involves quantifying various aspects of a gesture into a feature vector that properly describes it, so as to minimize misrecognition. Gesture recognition is usually performed in online systems i.e. a gesture must be recognized immediately following input. As a result, many features proposed for recognition are based on the actual handwriting of a gesture, which is extremely relevant in our context. Hence, in the following review we restrict ourselves to features that are calculated based on the handwriting information of characters. As a pioneering work in this area, Rubine (1991) proposes a set of 14 features for the purpose of pattern recognition in gesture recognition systems. Though these are constructed primarily for machine learning, he does observe that the features have been constructed in a way that could also be utilized to quantify user behavior. Long Jr et al. (2000) similarly define 22 features expanding on the original set proposed by Rubine (1991). Willems and Niels (2008) perform a very elaborate literature review on different types of features and propose several new ones of their own. They elaborate on a total of 90 different features for online single-stroke gesture recognition. This entire feature set was later distilled to 49 base features that are optimal for online symbol recognition (Delaye & Anquetil, 2013). Willems et al. (2009) also suggest additional features that pertain to multi-stroke pattern recognition. It can be seen that there are indeed a plethora of features,

but many of them are not particularly aimed at quantifying any specific property of characters (in terms of handwriting) and most do not provide substantial qualitative underpinnings for those features. They are mostly proposed as pure statistical descriptors to construct feature vectors for pattern recognition systems. However, there have been some preliminary applications of these features for semantic analysis. Long Jr et al. (2000) use their proposed features to analyze the subjective similarity of the gestures, but they do it indirectly using multi-dimensional scaling (MDS). Similarly, Vatavu et al. (2011) use some of the features to correlate with *perceived execution difficulty* of gestures. Though the above features were aimed at pen gestures, the features that closely correspond to the physical attributes can very well be adoptable for descriptive paleography as well.

In the next section, we will see various techniques that can mathematically model the production of handwriting.

2.3 Handwriting Models

As a human-oriented skill, handwriting has been a focus of interest for various fields such as psychology, neurology, forensics and computer science, with each emphasizing a different aspect of handwriting. Being a fundamental component of human motor control and also being one of the modes of interaction with digital devices, it is particularly important from a technical perspective. Handwriting is a very intricate activity that is produced by the complex coordination of cognitive, neural and muscular systems. Apart from physiological factors, external physical factors such as writing instruments and materials also affect the handwriting process, making it more complex to study. However, modeling handwriting generation can provide us with a basic understanding of various processes that are involved in the production of handwriting and more importantly also their interactions. This is very helpful to improve methods/features that primarily depend on handwriting such as online OCRs, gesture recognition systems and in our specific case, quantitative paleographic systems. There are several paradigms available for modeling of handwriting, and these can be generally categorized into

movement simulation methods and shape simulation methods (Dolinskỳ & Takagi, 2007).

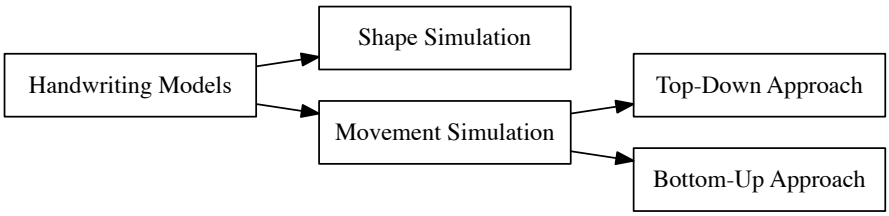


Figure 2.1: Categorization of handwriting models

Shape-simulation techniques do not require any of the dynamics associated with handwriting movement. They only consider the static shape of the generated handwritten character. Wang et al. (2002) propose a generative model that is based on control-points and B-splines. They train a model using handwriting samples and extract training vectors, which are then used to synthesize trajectories of whole words. In a similar way, Choi et al. (2003) use Bayesian networks to train handwriting samples. The shape is generated by searching for the most probable input point sequences. Xu et al. (2005) create aesthetic Chinese calligraphy through machine learning by additionally incorporating geometric constraints that reject unaesthetic shapes. Most of these techniques include using statistical methods to generate handwriting through (machine) learning from pre-existing samples. While this is an efficient paradigm to generate handwriting, it does not throw any light on the process of generation.

On the other hand, movement simulation is based on the motor models of handwriting. It attempts to model the actual physical process of handwriting (Wang et al., 2005). This method of modeling handwriting as a bio-mechanical process has two distinct approaches - top-down and bottom-up. The top-down approach is more concerned with the neurological and neuro-muscular interactions that generate handwriting, which is of less interest to us. However, the bottom-up approach is focused on the actual process at the hand-paper interface. In this, the handwriting trajectory motion is analyzed and modeled upon actual physical para-

meters such as velocity, force, pressure, spatial target, etc. (Plamondon & Maarse, 1989). Many models have been proposed under this bottom-up approach.

Hollerbach (1981) suggests a continuous model for handwriting generation consisting of two orthogonal sinusoidal oscillatory moments (horizontal and vertical), which are superimposed horizontally with a constant velocity. The oscillations are responsible for the shape of characters, while the horizontal sweep motion combines them. The model is given by parameters such as horizontal/vertical velocity amplitude, horizontal/vertical frequencies, horizontal/vertical phases, horizontal sweep and amplitude of motion. However, the model is mathematically complex and also not very intuitive. Singer and Tishby (1994) also propose a similar oscillatory model, but based on cycloid motion. Other models depending on orthogonal muscle movements can be seen in Denier and Thuring (1965), Eden (1968), and Koster and Vredenbregt (1971).

Morasso and Ivaldi (1982) detail a computational model of generating handwriting that is more intuitive. Here, handwriting is considered to be composed of basic curve elements called *strokes*, which overlap during production to form a visible trajectory i.e. shape of a character. The curves, which are represented as polynomial segments, are composed piecewise as a weighted sum over time to produce a smooth trajectory. In this way, the strokes are effectively *hidden* and are not immediately discernible. He also provides parameterized representations of rectilinear and circular strokes. While the model is very intuitive and relatable compared to the previous ones, it is far too simplistic, especially regarding the composition of strokes. The parameters of the model are partially geometric and cannot be directly related to the handwriting process. Improving upon Morasso and Ivaldi (1982), Edelman and Flash (1987) also decompose handwriting into strokes but they identify four basic stroke types - hook, cup, gamma and oval - that make up handwriting. The shape of a character is to be composed of these basic strokes. The kinematics of handwriting is then inferred from the shape. Though the model is an improvement, it is also still very geometric, and kinematics derived from shapes are not straightforward to parameterize. Bezine et al. (2004) describe a beta-elliptic model of handwriting as a superimposition

of elliptic stroke primitives, which are defined as mathematically complex beta functions. We see a clear convergence of ideas here, concerning handwriting being composed of primitive segments called strokes.

Of particular significance to us is the kinematic theory of rapid hand movements proposed by Plamondon (1995). It describes human handwriting as composed of strokes, which have asymmetric bell-shaped velocity profiles that are represented through log-normal functions. This closely mirrors the actual process, where an (ideal) handwriting is characterized by multiple bell-shaped velocity profiles. The kinematic theory offers a family of hierarchal handwriting models (Plamondon & Djioua, 2006) such as the Delta-Lognormal model and the Sigma-Lognormal model. The former can only predict rectilinear strokes, whereas the latter can predict complex curvilinear strokes as well. Since characters are made up of such strokes, Sigma-Lognormal is suitable for modeling actual character shapes. Under this model, each stroke is represented as a vector of six parameters. The handwriting trajectory is finally represented as a vectorial sum of all the constituent strokes. A stroke s is given by:

$$s = f(D, t_0, \theta_s, \theta_e, \mu, \sigma) \tag{2.1}$$

D is the amplitude of the stroke, t_0 is the initiation time of the stroke, θ_s and θ_e are starting and ending angles of the stroke, and μ and σ are neuro-muscular parameters. These parameters can be used to manipulate the shape of an individual stroke and also to control the amount of overlapping with a succeeding stroke.

The Sigma-Lognormal model is a widely popular handwriting modeling technique used in a variety of contexts. Djioua et al. (2006) developed an interactive tool that constructs a Sigma-Lognormal model for a given character (based on its trajectory). The tool then allows users to vary the parameters, which in turn affects the shape of a character. They propose that by varying the parameters and hence observing/using the shape change, we can perform multiple tasks such as studying outlier signature specimens, analyzing the qualities of handwriting forgeries, etc. In the same vein, Djioua and Plamondon (2008) use the model and a similar tool to generate unlimited samples of handwriting from a handful of

handwriting specimens. Here, the parameters of the Sigma-Lognormal model are varied to generate additional specimens as required. They propose that these synthetic data can be used to train and test online handwriting classifiers. Almaksour et al. (2011) use a similar method to increase and improve handwriting classifiers through synthetic gestures derived from deforming the model gestures. Ramaiah et al. (2014) take advantage of the model to add distortions to handwritten text, which are then employed as CAPTCHA. It is also used to generate and analyze graffiti tags by Berio and Leymarie (2015). According to them, this model allows the production of curves that are very similar, both visually and kinetically, to those made by humans using modern implements of writing. One of the reasons for such a widespread usage of this model is its extreme simplicity with only six parameters as we have seen. This makes it not only very effective but also an elegant way to computationally model human handwriting movements. We will be using this technique in particular for our further analysis. It will be expanded and explained in detail in the next chapter.

2.4 Summary

We began by summarizing the related research in the field of digital paleography. We identified the gaps and drawbacks present in the current digital paleographic systems, which would be addressed through our research. We then proceeded to describe the background concepts that are required in the context of our research relating to features of characters and modeling of handwriting production. In the next chapter, we will present the main methodologies underlying our work.

Chapter 3

Quantitative Analysis of Scribal Handwriting: Methods and Metrics

> Writing, Phaedrus, has this strange quality, and is very like painting; for the creatures of painting stand like living beings, but if one asks them a question, they preserve a solemn silence.
>
> Socrates to Phaedrus, an Athenian aristocrat[1]

3.1 Kinematics of Handwriting

Handwriting is one of the key themes driving the research presented in this thesis. It is, therefore, necessary that we first elaborate on handwriting and its production before we present and discuss our work.

Handwriting is a dynamic process that is produced by the movement of an implement on a surface. Even though writing a character is often seen as a single contiguous hand movement, it is actually made up of several sub-movements. It is a fluid process, in which these movements

[1] From *Plato's Phaedrus* (Fowler, 1925)

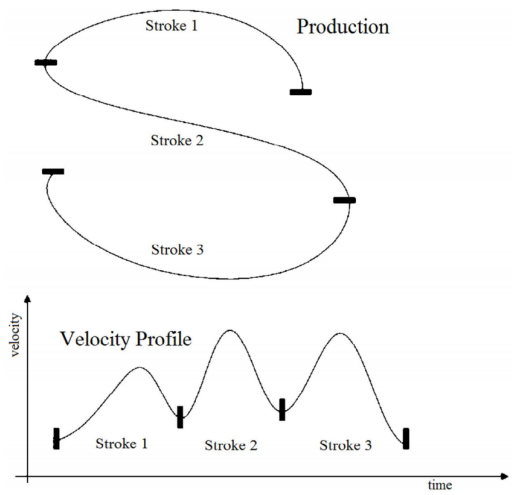


Figure 3.1: Velocity profile of the character *S*

overlap and compose to form a character (Morasso, 1981). The overall movement of an implement across the writing surface to produce a character is called a *trajectory*. Handwriting, being dynamic, can be described in terms of physical parameters such as velocity and acceleration with respect to an implement's movement. It is usually categorized as a ballistic activity, where sub-movements that each exhibit three distinct phases - an acceleration phase, a velocity phase and finally a deceleration phase (Teulings & Schomaker, 1993). During such a sub-movement, an implement initially accelerates until it comes close to the mid-point of that movement, after which its velocity stabilizes for a very brief moment. This is followed by a deceleration, where the velocity decreases as it reaches the end point. If it is just a single such sub-movement in isolation, the implement comes to a complete halt. Otherwise, it is followed by another acceleration phase for the next consecutive sub-movement. This behavior is understandable as an implement has to accelerate to reach its target and then decelerate gradually as it nears the target. This results in a characteristic bell-shaped velocity profile with a distinct peak, which typically occurs during an ideal, smooth and uninterrupted writing process. The sub-movement which corresponds to a single bell-shaped velocity profile we call *primitive stroke*. When there are several primitive strokes concatenated together, which we call *composite strokes*,

it results in a velocity profile with several peaks each corresponding to an individual primitive stroke. Thus, the writing of a character can be kinematically represented as consisting of several contiguous bell-shaped velocity profiles. Figure 3.1 shows the velocity profile of the character *S* (as written by a modern implement) with three distinct bell-shaped peaks each corresponding to a primitive stroke.

The moment when an implement touches a surface is called a *pen-down* event and the moment when it leaves the surface is a *pen-up* event. When a character requires only a pen-up and pen-down event pair it is called a *unistroke character*. For instance, Latin letter *S* is a unistroke character (in modern writing), where the lifting of the implement occurs only once i.e. when the writing is complete. But in some cases, an implement has to make multiple discrete contacts with a surface. These are called *multistroke characters*. For example, writing Latin letter *t* requires the implement to leave the surface at the end of the vertical stroke and then touch the surface again to complete the horizontal stroke. Thus, it consists of two pairs of pen-up and pen-down events. The term *stroke* here refers to the overall movement of an implement in continuous contact with a surface i.e. between a pen-up and a pen-down event (which we later specifically refer to as *pen-strokes*). The movement of an implement between two consecutive pen-up and pen-down events in multistroke characters is called a *pen-drag*.

3.2 Digital Paleographic Framework for Quantitative Analysis of Handwriting

As discussed in the previous chapter, while several digital paleographic systems are aimed at analyzing characters, none of them are particularly interested in analyzing scribal behavior quantitatively for descriptive paleography. As a result, more often than not, they do not have an established rigorous way to study characters' shapes. They analyze the shapes in terms of a collection of pixels on a screen, which we think is not appropriate for descriptive analysis. We argue that if we are to analyze characters systematically, we need a suitable computational framework that operates on a proper paradigm. This requires having a theoretical appreciation of the underlying handwriting processes. To elaborate, we have to understand the processes behind stroke creation and interaction that define the corresponding scribal behavior. The paradigm we choose is that of the *ductus* feature of characters.

Below, we propose a framework that considers the handwritten motion of characters as a fundamental property and operates based on that paradigm. This serves as the theoretical guiding factor for the entire framework. Analyzing characters based on how they are written is also a very intuitive way to look at them. The proposed framework can be applied for both quantitative and descriptive analysis. The quantitative metrics obtained from the framework allows an expert to access a wide variety of statistical methods that can be applied in the context of analyzing scribal behavior. It thus facilitates them to perform innovative and interesting analyses, which usually cannot be done with qualitative methods in a descriptive context. At a general level, it is to be used for studying and comparing handwriting behavior of various scripts/scribes and also to understand why a particular script/scribe has a specific feature. These applications would be extremely useful in the context of descriptive paleographic analysis. We also expect the framework to be useful to understand the nature of human handwriting, and to discover if there are specific features or patterns when humans write. With respect to qualitative analysis, it can also aid it by providing effective quantitative

support.

Most of the modules proposed in the framework are computational and therefore can be largely automated. While one of the motivating factors of the framework is automated analysis, we do recognize the role of expert users interacting with our system. We are of the opinion that they should be able to inject their knowledge as required, as it can enhance computational processes. Hassner et al. (2014) argue for a human-oriented approach in digital paleography and note, "'human in the loop' can and should be integrated into all stages" to "overcome the shortcomings of strictly automatic approaches". Accordingly, we incorporate user input into our framework and allow users to override and perform manual operations as well. It will be seen that the framework, in fact, provides various avenues for interaction with each step of its processing. Effectively, this results in a semi-automatic approach that can be augmented with human judgments as required.

Such a human-aided approach is, as matter of fact, better suited to dealing with paleographic scripts, which frequently require reconstructions and as a result also frequent subjective decision making. While we do not aim to completely eliminate human subjectivity from the process, we attempt to streamline the amount of subjectivity involved by making the underlying process more explicit. This allows users to interact with the system within our paradigm at a level comfortable to them. They are usually quite wary of using entirely automated approaches and are not completely convinced to trust the output of a given software that performs a task that has traditionally been performed manually. The proposed framework is not designed to be used as a black-box application, where a user imports characters only for the framework to expunge a collection of opaque numerical values. It is intended to be used as a *gray box*, where users can see the principles and guiding factors behind each analysis and interact with them. As noted by Stokes (2007), any ad-hoc involvement by the user to manipulate and/or improve the results must be logged for the sake of reproducibility. Any implementation of this framework should include this feature.

The framework tries to follow some of the suggestions made by Stokes (2009a) for a successful digital paleographic system. It aims to

be *reproducible* as it is strongly grounded on the theoretical principles of handwriting generation. This also makes the results of the framework more *interpretable* or *communicable* as they can be discussed directly in terms of handwriting behavior with respect to specific scribes and/or specific characters. The framework is also *flexible*, in that any change in the underlying assumptions can be comfortably incorporated with minimal overhead. Overall, it aims to provide an overarching *common framework* to quantitatively analyze characters for descriptive paleography, through the handwriting information of characters. Within this broad context, the framework is very open-ended, extendable and customizable to any particular situation. It is also highly *modular* with its modules being self-contained with the outputs of the preceding modules being the inputs for the succeeding modules, and their results can, therefore, be improved or edited as and when required.

In the following sections, we discuss the individual modules contained in the framework. We explain in detail their motivation, inputs, workings, outputs and also the associated limitations/assumptions (if any).

3.2.1 Spline Representation

The first module of the framework pertains to the initial digital representation of characters. As seen in section 3.2, many paleographic systems use pixel representations for their analyses as they frequently employ image-based techniques. Though simple and convenient, they are not ideal for analyzing characters in terms of handwriting production. Therefore, we attempt to find a suitable representation for characters in the context of our analysis.

In the field of computer-aided design (CAD), mathematical representations called *splines* are often used to represent complex shapes. These are parametrized representations, which model a complex curve by reducing it to a set of points that represent its shape. They are mathematically simple, easy to manipulate, and relatively easy to implement and use as well. Spline-based representations are considered for representing handwritten characters by Morasso and Ivaldi (1982), who use them for their analysis. We propose that such a spline-based representation is both

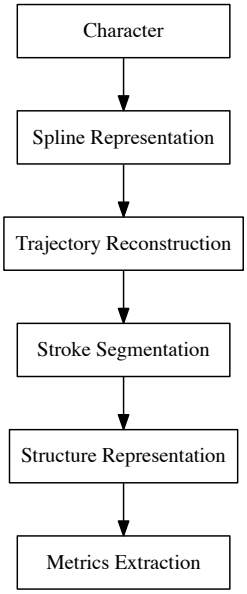


Figure 3.2: Modules of the framework

natural and suitable for our processing as well.

There are several types of splines and of particular interest are *Bezier splines* and *B-splines*. Bezier splines are the most widely used because of their computational simplicity. For instance, most fonts use them to represent the shapes of their glyphs. A font glyph is essentially a composition of several Bezier splines that define its shape. While Bezier splines are indeed simple, they are not scalable. As curves become larger (and more complex), it becomes much more difficult to manipulate them as they do not provide local shape control. It is hard to modify a particular curve segment without affecting other parts. Moreover, to represent more complex curves we have to resort to Bezier splines of higher order, which are computationally more complex and even harder to manipulate. Given that we seek to approximate a character for any size and shape, this is particularly disadvantageous to us. Instead, we turn our attention to B-splines (De Boor, 1978). They are very similar to Bezier curves but are simpler. They consist of a number of sub-segments (called piecewise polynomials), which make up the spline. Hence, they distinctly

offer the option of localized shape control. Any changes made to a curve segment is localized to that particular piecewise polynomial. To represent longer curves, a larger number of piecewise polynomials is used without resorting to higher-order representations. B-splines, like Bezier splines, also have control points that can be manipulated without significant effort. For these reasons, we computationally represent characters using B-splines.

Conversion of characters' shapes into their B-spline based representation can be done either manually or automatically. In the former case, the shapes of characters are explicitly constructed using a set of B-splines, with users defining them for their analysis. But very often, characters already have existing digital representations that are image-based. Hence, we need to provide a way to automatically convert image representations to the required B-spline representations. This is done in multiple stages. In the first stage, corners of the imported images are detected using a robust standard corner detection algorithm (Chen, Zou, Zhang & Dou, 2009) such as the Harris operator. Once they are detected, we attempt to find and list all connected pixels between all the adjacent pairs of corners. Using a standard curve reduction algorithm, these lists are reduced to the bare minimum required to capture the shape of curve segments between the corners. We find the Douglas-Peucker algorithm (Douglas & Peucker, 1973) to be particularly effective in performing this task. These reduced lists are then used to create the corresponding B-splines through standard spline interpolation. This finally results in a spline representation of a character, where the constituent curve segments are now B-splines. Figures 3.3 and 3.4 shows sample results of spline conversion. When we encounter loops in a character, we insert pseudo-nodes to facilitate trajectory reconstruction (which will be discussed in the next section). In figure 3.4, nodes F and G are essentially pseudo-nodes.

While B-spline representations may be ideal in most cases, they have some limitations. Using plain B-splines results in the loss of certain information such as stroke thickness and angle of instrument that are important in certain paleographic contexts. If such meta-information is found to be fundamental to a script, the data structure that holds B-splines needs to be augmented with additional attributes as required.

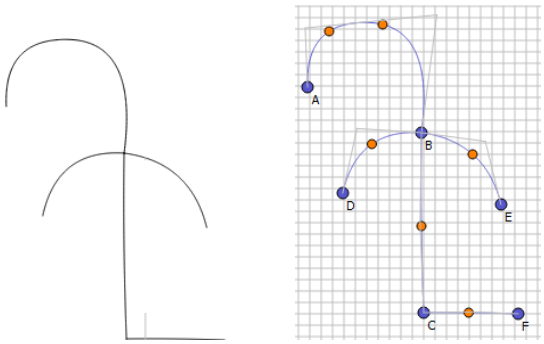


Figure 3.3: Spline representation (right) of a character (left)

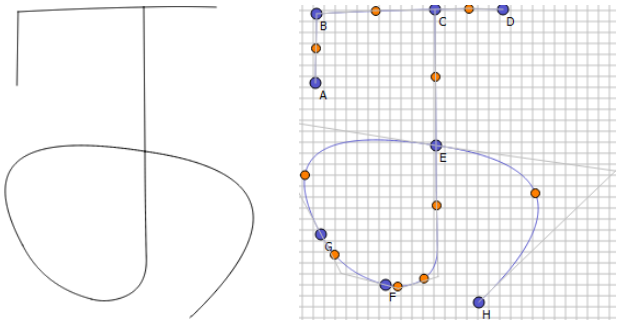


Figure 3.4: Spline representation of a character with pseudo-nodes on loops

As such, the framework leaves open the ability to augment additional information.

3.2.2 Trajectory Reconstruction

Using spline representations we only capture characters' shapes, to be more precise, their static appearance. They do not (yet) contain any information about production, i.e. trajectories, which is required to properly analyze characters. The second module of the framework pertains to the extraction of trajectories from spline representations.

For contemporary scripts, trajectories are often well known. In that case, users can directly overlay them on characters. However, we are more concerned with paleographic scripts, whose trajectories are often unknown. We propose that for characters of such scripts, we recover their trajectories. Blanchard (1999) discusses this difficulty in recovery and

notes paleographic artifacts usually do not preserve such information. However, based on the static shape of characters it is possible to perform a reasonable reconstruction of their trajectories. Doermann and Rosenfeld (1995) suggest that the recovery can be obtained by:

1. Global cues such as

   Relative direction of handwriting

   Minimizing of effort/energy required for production

2. Local cues such as:

   Striations

   Stroke width variations

We are more focused on a high-level reconstruction and therefore are more interested in global cues rather than local cues. We feel that global cues provide a generic abstraction about writing characters. Trajectory recovery techniques may be classified into three approaches. The first approach is the graph theoretic approach, as suggested by Bunke et al. (1997), Jäger (1996) and others, which performs the trajectory search on a graph. In the second approach suggested by Doermann and Rosenfeld (1995), Lee and Pan (1992), and Lallican and Viard-Gaudin (1997), the search is performed on the image contour or skeleton of an image and typically includes local cues to aid the recovery. In the third approach, seen in the work of Lau et al. (2003) and Nel et al. (2005), recovery systems are typically trained with online data that is then used to recover the information from a test set. Nguyen and Blumenstein (2010) have made a comprehensive survey of various techniques for trajectory reconstruction.

In our context, the machine learning approach is not applicable as we do not have existing trajectory data to begin with. The image-based approach as it stands focuses more on local cues and hence is also not very suitable for our purpose, as we prefer a higher level approach to recovery. Based on our need for a global approach, graph-based methods are well suited for us.

We are particularly interested in the work of Jäger (1996), who proposes a simple but efficient method based on Eüler's path generation. In

the case of paleographic scripts, where original trajectories are often not known, there is not a unique answer. Very fittingly, his work is able to provide several alternative theoretical trajectories ranked according to their viability. It also facilitates imposing several additional heuristics specific to a script as the recovery is performed using high-level handwriting behavior as desired. In Jäger's method, a character is mapped into a graph and Eülerian paths for that graph are generated. These paths are then ranked based on the length minimization and curvature minimization principles. Usually, humans tend to follow shorter paths (i.e. length minimization) and write smoother strokes with less deviation (i.e. curvature minimization), as opposed to longer paths with rugged strokes. It is assumed that an ideal trajectory, in this way, attempts to minimize the effort to produce a character. We have adapted Jäger's algorithm to fit our purpose. We elucidate below the modified algorithm for recovering trajectories.

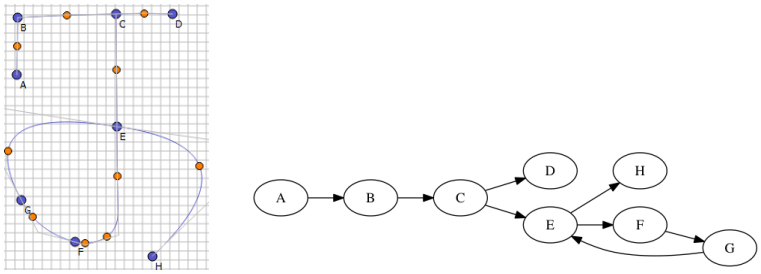


Figure 3.5: Graph representation of a character

We begin by abstracting the spline representation into a higher level graph representation. Each node in the graph represents a corner, and the B-spline segments are represented as edges. The data structure for the edges holds the corresponding B-spline segments. We then enumerate all possible paths of traversal available in the given graph, which is done by calculating the Eülerian paths for the graph (Ore & Ore, 1962). A path required to visit all edges in a graph at least once is called an *Eülerian path*. In terms of trajectory, this translates into the pen movements required to trace the entire shape of a character. We proceed to calculate various *costs of writing* for each generated Eülerian path, with respect to their global cues - length, curvature, and direction of writing. We calculate

a normalized score by assigning weights to each of these costs. This normalized score indirectly corresponds to the effort required to write a character. The top n paths are ranked according to their score and presented to users. We illustrate this with the following example.

Assume we have the following path p for the character in figure 3.5: A $\rightarrow$ B $\rightarrow$ C $\rightarrow$ D $\rightarrow$ C $\rightarrow$ E $\rightarrow$ F $\rightarrow$ G $\rightarrow$ E $\rightarrow$ H

We calculate the length cost of the path, $len(p)$, by summing the length of all edges i.e. B-spline segments in that path.

$$len(p) = len(A,B) + len(B,C) + ... + len(E,H) \tag{3.1}$$

For path curvature, we calculate the absolute value of the angle between successive edges, which is computed by calculating the angle between tangents of the two curves at the point of intersection. In figure 3.5, edge pair (G,E) and (E,H) have smooth transition and hence lower cost, compared to edge pair (D,C) and (C,E) which requires a sharp turn. The curvature cost, $curv(p)$, is the sum of all the angles covered when writing the path.

$$curv(p) = curv((A,B),(B,C)) + curv((B,C),(C,D)) + ... + curv((G,E),(E,H)) \tag{3.2}$$

We also calculate the directional cost by invoking common heuristics. Most scripts have a preferential direction of writing such as left to right or top to bottom. For instance, if we are sure that the usual direction of a script is left to right and top to bottom, then trajectories following these directions are given higher priority by penalizing other paths. We do this by assigning negative scores to such paths. The heuristics have to be decided based on the script under consideration. $dir(p)$ would then give the directional cost for the path.

The total cost for the path is calculated by:

$$cost(p) = w_1 \cdot len(p) + w_2 \cdot curv(p) + w_3 \cdot dir(p) \tag{3.3}$$

The weights w_1, w_2 and w_3 are fractions that sum to 1 and are assigned empirically based on the script under consideration. The relative weights of 0.4, 0.4 and 0.2 respectively were found to be appropriate for Indic

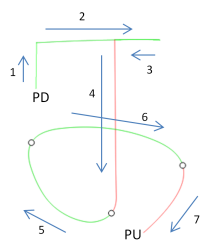


Figure 3.6: Reconstructed trajectory of character. PU and PD are pen-up and pen-down events respectively

scripts. The individual costs are normalized between 0 and 1 for this calculation.

From the returned top n paths, users can select a trajectory of their choice. However, during this selection they also need to consider other external factors such as consistency with trajectories of other characters in the same script, allographic variations, etc. See figure 3.6 for the reconstructed trajectory of the character in figure 3.5. In some cases, particularly with calligraphic scripts, the basic assumption of reconstruction, namely *effort reduction*, might not hold at all. In such cases, augmentation of users' practical external knowledge particularly enhances this process. In such cases, they can override the suggested trajectory by either partially modifying it or just replacing it with their own choice. Also, the heuristics used can always be customized to a specific script under consideration. For multistroke characters, the trajectories must be reconstructed for each pen-stroke and then they should be ordered based on a higher level heuristics to recover the overall path. For instance, a longer pen-stroke is most likely to be written first followed by shorter ones.

To summarize, we take a spline representation and abstract it into a higher level graph representation. Using this higher level graph representation we attempt to recover trajectories by applying the heuristics of effort minimization. At this stage, we have recovered the fundamental information pertaining to a character. We can now proceed to the next module of the framework, where we actually start to analyze the character.

3.2.3 Stroke Segmentation

A precursor to performing any kind of analysis on a structure would be to decompose it into its fundamental constituent parts. Such decomposition provides a finer view of the structure and gives insight into its construction, and also relations among its constituent parts. Similarly, to analyze a character we propose that it also needs to be decomposed into its fundamental parts i.e. strokes. Stansbury (2009) supports this argument by stating that "analyzing letterforms into their component strokes and pen angles" is a fruitful approach for digital analysis. Bishop (1961) further elaborates that "[e]ven more than in ductus and sense of form and proportion the idiosyncratic is to be found in the production of single strokes, in the behaviour of the pen as it turns a curve or a corner [...]" (Stokes, 2009a). This reiterates the importance of extracting strokes from characters using a proper underlying principle.

There have been several approaches to decompose characters. Edelman and Flash (1987) decompose characters into four different templates - hook, cup, gamma and oval. We feel such predefined decomposition is not suitable for the creation of proper primitives required for our analysis. Writing is a natural process, which cannot be simply reduced to a set of predefined templates. Changizi et al. (2006) decompose the characters into *separable strokes* using three subjects who decide (unanimously) on the decomposition. Such a completely subjective process relies on some underlying criteria that are often unknown and do not directly correspond to the handwriting process. A better alternative is to have a specific process as a guiding factor to perform character decompositions, which helps in automation, and at the same time, it also provides a reasonable set of guidelines through which users may choose to interact with the process. In our particular context, the primitives of characters would be the fundamental strokes involved in their production. This is consistent with the way they are internalized and produced by humans. Based on our chosen paradigm, we propose to use handwriting information (as reconstructed by the previously discussed module) to decompose characters into their fundamental parts.

Before we proceed, we have to perform the restructuring of characters'

spline representations. The glyphic segments generated do not directly correspond to the actual strokes of characters. Therefore, using the recovered trajectory, characters are reconstituted as a set of B-splines representing the overall composite strokes that directly correspond to their trajectories. For instance, if two curve segments are part of the same smooth stroke, they are combined into a single stroke. At this point, we have already decomposed the character into composite strokes. But we have to further decompose these to extract primitive strokes. Figure 3.7 demonstrates the restructuring of the character shown in figure 3.3. Note how the glyph segments have been combined to form composite strokes.

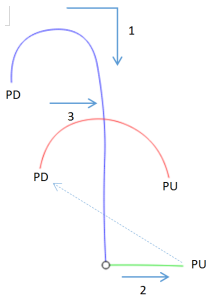


Figure 3.7: Character with composite strokes as a result of restructuring .

With the reconstituted representation of characters, we now attempt to retrieve the primitives. This retrieval is performed by segmenting the trajectory at appropriate points. Hence, the process is appropriately named *stroke segmentation*. As expressed in section 3.1, writing a character is not a discrete, but a continuous process where individual strokes overlap. Based on a character's trajectory, we proceed to find specific points where the (apparent) primitive strokes connect. It has been shown that the minimal velocity points occur where the curvature is maximal or minimal and also where strokes are explicitly delineated such as at sharp junctions (Li, Parizeau & Plamondon, 1998). When the composite strokes are segmented at sharp junctions, we extract what are termed *disjoint strokes*. Figure 3.7 shows the various disjoint strokes in different colors. The extreme points of curvature to extract primitive strokes are automatically detected from the disjoint strokes. This is particularly simple in our case, where the spline based representation very easily

yields such calculations. The x and y coordinates of a B-spline are given as a function of a parameter t.

$$x = x(t) \quad y = y(t) \tag{3.4}$$

The curvature k at each point of a B-spline is calculated using the derivatives of the above parametric functions.

$$k = \frac{x'y'' - y'x''}{(x'^2 + y'^2)^{\frac{3}{2}}} \tag{3.5}$$

where the prime and double prime refer to the first derivative and the second derivative respectively.

Using this equation, we calculate the curvature at each point of a disjoint stroke and then attempt to find the local maxima and minima to extract the segmentation points. One must notice that this method is extremely sensitive and can detect even very minor changes to curvature. To overcome this, we impose additional heuristics to appropriately filter them. Initially, we set an empirical threshold for curvature - only points above this are considered to be segmentation points. Additionally, if the distances between two or more of these points are less than a threshold, they are combined. The disjoint strokes are then segmented at all these points where the primitive strokes overlap and connect. See figures 3.8 and 3.6 for illustration. These segmented primitive strokes are a good approximation of the underlying strokes. In this way, we produce a natural set of primitives corresponding to a character. For a multistroke character, the pen-drag between the individual strokes is included as an additional invisible stroke because it also involves a movement of hands. It may also be necessary to override the automated stroke segmentation process on a case-by-case basis. For instance, given a trajectory, it might be more practical to have one longer stroke instead of two successive short strokes.

During stroke segmentation, we assume that a writing implement has a smooth, unhindered movement over a writing medium, which results in a bell shaped velocity profile for any given primitive stroke. Even though this is mostly true for many forms of writing (both modern and ancient), in the case of a movement quite different from our assumption

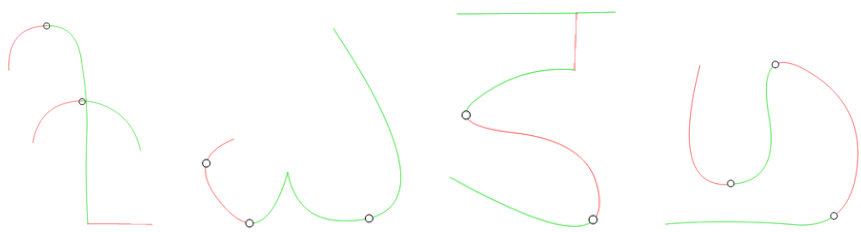


Figure 3.8: Disjoint strokes decomposed into primitive strokes. The different colors refer to up-strokes (red) and down-strokes (green). See §3.2.4.

the definition of stroke needs to be redefined to suit the specific physical process that produces the writing under consideration.

3.2.4 Structure Representation

Following the decomposition of a character into its primitives, we create a hierarchical structure based on the results of decomposition to represent the character.

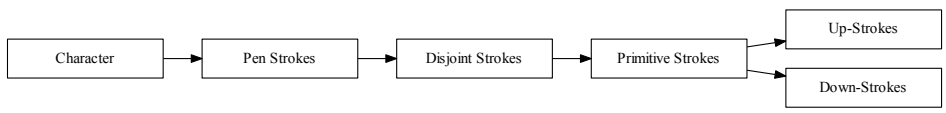


Figure 3.9: Stroke hierarchy

A stroke hierarchy is first constructed as shown in figure 3.9. At the top are *pen strokes*, which are the total movements written without lifting the pen. The multi-stroke character illustrated in figure 3.7 has two pen strokes. These pen strokes can be divided into what we call *disjoint strokes* that are delineated by sharp junctions, as seen in the previous section. Disjoint strokes are further composed of *primitive strokes*, which are again further classified into up-strokes and down-strokes. These possess two different characteristics with completely different physiological processes of production. It has been shown that up-strokes are susceptible to change, while down-strokes are invariant (Teulings & Schomaker, 1993)

and more stable (Maarse & Thomassen, 1983). Up-strokes are faster to produce (Isokoski, 2001), which may also lead to their lower stability. Maarse and Thomassen (1983) defines strokes that are produced between 210° and 280° to be down-strokes. The range of angles appears to be very restrictive (as it considers only Roman handwriting). Hence, we have included strokes that are pointed downwards within 210° and 330° as down-strokes and all others as up-strokes. The criteria to judge up-strokes and down-strokes should be modified (or even inverted) based on the writing style and/or writing implement.

The structural representation of a character is then built based on our proposed hierarchy, which can abstract characters at any stroke level as needed. In this new representation, a character is composed of strokes rather than curve segments. They are also represented as B-splines similar to the curve segments of a character. Such a fundamental representation using primitives derives quantitative features that are more descriptive. This representation can be used for other kinds of related analyses as well (see §3.3, specifically §3.3.3). If necessary, a structured and detailed XML representation as suggested by Terras and Robertson (2004) could also be adopted.

Additionally, this representation can be used to view the stroke inventory for a script. This inventory can be abstracted at any stroke level suited to an analysis. To create the script inventory, we collate all the individual strokes and reduce the inventory list by merging similar strokes based on a cost threshold that is empirically chosen. This can be particularly useful if the motivation is to study patterns appearing in scripts and how they change and evolve.

3.2.5 Metrics Extraction

One of the main purposes of this framework is to extract metrics that describe characters to facilitate easy and effective quantitative analysis of scribal behavior. We use the term *metrics* here synonymously with *features* but we think the former conveys the connotation of quantifying information better. Many features used among digital paleographic systems, as described in section 2.2, are more focused towards the visual

aspect of a character, and being highly statistical, are more suited to machine recognition. Such metrics cannot always be correlated with some explicit qualitative features as perceived by humans. Our stroke-based representation is much more elegant and apt, capturing both the visual and kinematic information. Through this, we attempt to find relevant metrics (with qualitative significance), to quantify scribal handwriting behavior. The metrics can serve as descriptors for scripts, and be used for comparing, analyzing and classifying them and hence fit well within the traditional context of paleography as well.

In the section below, we discuss the selected metrics from our review of features as discussed in section 2.2, but we also propose new metrics that we think are useful and relevant to us. We carefully study each of the features and choose only those that have qualitative real-world significance and are directly related to handwriting production of characters. We reject very abstract mathematical features that do not have any direct qualitative significance. Many of the rejected features were artificially created as part of *feature construction* to increase the number of features in a feature vector, which is a common process in the field of pattern recognition. This process is usually performed by applying mathematical functions such as logarithms, sines and cosines on other features, which cannot be interpreted easily. There are also features that are very opaque and statistical such as Hu-moments, eccentricity, perpendicularity and cannot be easily correlated to a real-world or semantic attribute about characters. There might be some overlap below with the paleographic differentiators as proposed by Mallon (1952) and others (see §2.2), as we attempt to quantify all relevant metrics arising from a character's trajectory. In the case of such overlap, we elaborate them in the context of our analysis.

We can divide our metrics into two overall categories - absolute and derived. Absolute metrics are calculated directly from the structure based on particular properties and are not scale invariant, e.g. length. Therefore, it may be required to normalize the metrics before using them between two different scripts/characters. Derived metrics are often ratios between two absolute metrics. The basic premise is that ratios capture information that is more helpful than an individual value. We extract many metrics

that we consider to be useful, however, the stroke structure allows the extraction to be very open-ended and a number of other metrics could also be proposed from the structure depending on our needs.

Many features as discussed below were originally designed for gesture recognition systems and therefore are intended for systems that either use a mouse or a stylus as an input device. Even though they assume modern digital implements, with a few exceptions, they are not dependent on the writing implement. They rely on the general process of human hand movements for writing. We may safely assume that, at a very fundamental level, handwriting activity remains the same irrespective of implements used. On that assumption, we attempt to explore the use of these metrics on medieval and ancient handwriting. If a metric is specifically dependent on an implement or a specific way of writing, we explicitly mention the accompanying assumptions and limitations.

We categorize below our metrics based on the type of information they intend to quantify.

Visual Information

Visual information directly pertains to the appearance of a character. The following metrics attempt to quantify the different aspects of it. These also partially quantify some production properties along with visual properties of a character. Many of the features proposed below roughly fall under the category of the paleographic differentiator, *Modulus* (see §2.2).

Length This is the total length of a character. In the case of unistroke characters, this is calculated as the sum of the individual primitive stroke lengths. For multistroke characters, this includes the sum of the primitive stroke lengths and also the movement during the pen-drag, which is approximated to a straight line. Thus, length quantifies the entire movement of an implement required to produce a character.

Divergence Divergence is defined as the distance between the position of the first pen-down event and the last pen-up event. This metric quanti-

fies the movement of an implement between those two events measuring how much the implement has visually *diverged* from its original starting position. This is one of the important metrics that could be specific to an individual scribe.

Figure 3.10: Divergence

Size Size is measured by the area of the bounding box of a character. The bounding box is the smallest rectangle that encloses a character (See figure 3.11). This could be directly correlated with the size of a character. A bigger bounding box corresponds to a larger character size.

Figure 3.11: Size

Length-Breadth Index This is the ratio of the height[2] of the bounding box to that of its width. This approximates the shape aspect of a character, e.g. slender, broad, etc.

Average Curvature This metric is calculated by averaging the curvature at all points of a character using equation 3.5 described in section 3.2.3. A straight stroke will have a curvature of zero, while a curved stroke will have a greater absolute value. Thus, curved characters tend to have a higher average curvature value compared to a character with fewer curves and/or more straight lines.

[2]The term *length* in Length-Breadth Index actually refers to the height. This is frequently used medical terminology and hence was adopted as such.

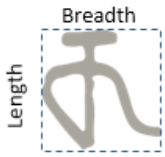


Figure 3.12: Length-Breadth Index

This partially corresponds to the *cursivation* features proposed by Burgers et al. (1995). However, our metrics can provide a bit more detailed information as we would be able to get curvature information per stroke as well.

Compactness The compactness of a character is defined as the ratio between its length and size. In some sense, it defines how compact (or dense) a character appears and indirectly corresponds to the length of strokes that a scribe is trying to fit within a given area. This makes it a very important metric to consider with characters, for it can be very specific to a scribe or script. Some scribes may space out a character during production while others may tend to *compact* the strokes within a small area. This could be helpful in the detection of hands in manuscripts.

Figure 3.13: Figure on the left is more compact than the figure on the right

Openness The openness of a character can be defined as the ratio between its divergence and length. This measures the movement of an implement with respect to its starting point and ending point, and the length of a character. The actual metric suggested by Long Jr et al. (2000) is the ratio of divergence to size. However, this does not appear to be ideal. We think it is better to compare two movements (divergence and length) rather than comparing movement to area (divergence and size).

Distinctiveness Several ways have been proposed to compare the appearance of characters. Mačutek (2008) proposes a very idiosyncratic way of calculating the distinctiveness of characters, which involves decomposing characters into basic templates and then comparing the permutations of the decomposed components. In several Optical Character Recognition (OCR) processes, pixel-based techniques, such as the image distortion model, are frequently employed to calculate the similarity between characters. Similarity (or lack thereof) is usually calculated using the cost of transformation between two entities. Entities possessing similar representations are readily transformed into one another, whereas dissimilar entities require many transformations (Hahn et al., 2003). Thus, the distinctiveness between characters is directly proportional to the transformations required to make them similar.

We propose to use the Dynamic Time Warping (DTW) distance (Müller, 2007) to calculate the distinctiveness between two characters. DTW is traditionally employed to compare two temporal sequences, which may vary in time or speed. DTW attempts to align two sequences and calculates the cost of the alignment. It has been widely applied to compare temporal sequences of audio and video data. DTW can be applied to any time series, hence it can naturally be applied to trajectory data as well. It has even been suggested for intuitive handwriting recognition (Niels & Vuurpijl, 2005). This makes it an ideal metric to measure for our purposes. Though aimed at trajectory data, DTW with suitable adjustments can also be applied to the static shape of characters. To apply DTW to a static shape, we linearize the image into a sequence from left to right and top to bottom, and create a pseudo-time sequence. This sequence is used for calculating the DTW cost for static shapes.

Ascendance and Descendance Some scripts have baselines and mean lines, and the portions of characters above these lines are called ascenders and descenders respectively. These can be used to derive some additional metrics. The percentages of the length of a character above and below the baseline are defined as its ascendance and descendance respectively. They were included in some of the paleographic differentiators that were proposed previously.

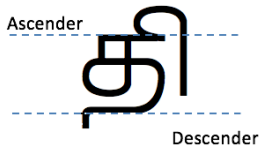


Figure 3.14: Ascendance and Descendance

Circularity and Rectangularity In many cases, characters appear to approach an ideal geometric shape. We attempt to measure such approximation. Circularity and rectangularity could be defined as the deviation of a character's outline shape from that of an ideal circle and rectangle respectively. For circularity, we take the ratio of the area of the convex hull (a polygon that encloses the outline shape) to the area of the minimal circle that encloses the character. Similarly, rectangularity can be calculated from the ratio between the area of the convex hull and that of the bounding box.

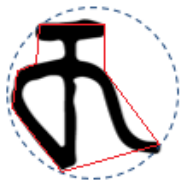


Figure 3.15: Circularity

Visual Complexity Visual complexity can be defined as the effort required to decode and recognize a given character (Köhler, 2008). Some characters are perceived as complex and others as simple. Altmann (2004) has proposed a technique in which a character is decomposed into lines, arches and curves with each component assigned a weight. The sum of these weights is calculated as the quantified complexity. Peust (2006) has proposed a complexity measure by counting the number of intersections that a character has with a straight line. These techniques do not appear to be rigorous and are not supported by any empirical studies. Similarly, using structural information theory (SIT) there have been proposals to quantify the *load* of a character. The higher the load, the more complex

the character is to be perceived. It involves measuring repeating patterns and assigning weights to angles of junctions (Hanssen et al., 1993). While SIT can easily work for simple geometrical shapes, extending them to complex shapes such as characters is very hard and not very practical. The methods described attempt to quantify a very abstract notion, which is more subjective than other metrics. People with exposure to different writing systems could quantify the complexity of a character in very different ways. Hence instead of aiming for complete quantification of character complexity, we propose to quantify only the factors (such as defined previously) that contribute to the visual appearance of a character. Using multidimensional techniques such as parallel coordinates, we could trace the change in factors that contribute to the visual appearance. Along with the previously listed factors, we can add factors such as the sum of inter-stroke angles and number of crossings, which may also contribute to visual complexity.

Kinematic Information

Apart from the visual information of a character we also need to consider its kinematics. The character's kinematic (or temporal) information is essential in defining it as it dictates how the character is produced through the process of handwriting. This is particularly useful in the context of descriptive paleography.

Stroke Counts A fundamental metric is the number of hand motions required to write characters. Humans generally attempt to minimize the number of hand-movements to write characters (Salomon, 2012), but in some cases, additional strokes are added into characters, increasing the production effort. It is an interesting metric to analyze for the distribution across various scripts. This also allows us to study human writing behavior by understanding the circumstances/environment under which stroke additions or reductions may occur. Apart from the count of the primitive strokes, there are two more composite-stroke metrics that could be considered - pen-strokes & disjoint strokes (see §3.2.4). For instance, figure 3.6 shows a character with one pen-stroke, three disjoint strokes

and eight primitive strokes. We could also include retraces in the count, where the same stroke is traced successively in the opposite direction. Movement 3 in figure 3.6 is a retracing stroke.

Stroke Length The distribution of the length of individual strokes and also the average stroke length are very useful measures with respect to the analysis of writing. The average stroke length is a variable entity across different scripts or scribes, and can be useful in classification. Stroke lengths are measured separately in terms of disjoint strokes and primitive strokes (up-strokes and down-strokes) as they quantify two different movements.

Changeability Handwriting strokes can be divided into up-strokes and down-strokes as seen earlier in section 3.2.4. The down-strokes are more stable than the up-strokes. Consequently, a character's tendency to change (with respect to stroke stability), i.e. changeability, can be estimated by the ratio of the length of up-strokes to that of the down-strokes. If it is high, the character can be considered susceptible to change.

Disfluency As described in section 3.1, writing is a ballistic activity. It is known that handwriting fluency is affected at points where the curvature is at its maximum/minimum. The number of sharp junctions in a character also contributes to the decrease of velocity during handwriting production. The count of all points that affect velocity is termed as disfluency. Based on our previously stated assumptions, these would be *curvature extrema*, *sharp-junctions*, and *intermediate pen-up events*. If any other events are known to affect the movement of an implement, they need to be included as disfluent points as well. This can indirectly correspond to the difficulty in terms of writing a character. A character with a higher number of disfluent points is harder to produce as the velocity is frequently interrupted. Similar measures have been used with actual handwriting velocity data to assess handwriting fluency of people by measuring the number of velocity inversions (Tucha et al., 2008). The character in figure 3.6 has 6 disfluent points.

In fact, the number of disjoint points can be taken as separate metric altogether, since its effect on slowing down the production is considerably higher than that of the other points.

Entropy In information theory, entropy is defined as the average amount of information contained within an entity. It is directly proportional to the randomness or disorderliness present in the system. When there are several instances of change in a system, it results in an increase of entropy as the system now contains more information (Aksentijevic & Gibson, 2012). To calculate the entropy of a character, its trajectory is *quantized* into chain codes (Freeman, 1974) denoting the major eight directions by assigning a chain code to each individual strokes. The eight chain codes correspond to the following directions - N, S, E, W, NE, NW, SE, and SW. For example, the sample character in figure 3.6 can be quantized into [N E W S NW SE SW].

Entropy is calculated with the following formula (Bhat & Hammond, 2009):

$$H(s) = -\sum p(s_i) \log_e p(s_i) \tag{3.6}$$

where $p(s_i)$ is the probability of a stroke, which is estimated by the ratio of the count of the given stroke (in a character) to that of the total number of strokes.

A character with a sufficient number of repeating strokes will record low entropy and those with fewer recurring strokes will record high entropy. Thus, the entropy of a character conveys the randomness associated with the hand movements required to produce that character.

N-Gram model of scripts Writing a character can be considered to be very similar to that of constructing a sentence. While sentences are made up of words, characters are made of strokes. Here, we seek to apply some aspects of natural language processing to scripts. N-gram modeling is frequently used in natural language processing for a wide variety of purposes. An N-gram model is a probabilistic model to predict the next item in a sequence based on the (N-1) previous items (Fink, 2014). The

model also provides an opportunity to derive several metrics. As the number of stroke combinations is usually low, a bigram model (N=2) is best suited to model script behavior. Accordingly, given a stroke (or a list of strokes), we can predict the most probable stroke that would follow it. This can be used to study the regularity of stroke combinations.

Assume a character c is a collection of n strokes, it can be represented as a chain code representation c^d.

$$c^d = \begin{pmatrix} w_s & s_1^d & s_2^d & s_3^d & \dots & s_n^d & w_e \end{pmatrix} \tag{3.7}$$

where w_s and w_e refer to the start and end of the writing process, and s_n^d refers to the direction chain code of the nth stroke as seen in the previous section. The bigram model assumes that a given stroke is dependent only on the previous stroke that was written. (In contrast, a trigram model would have assumed a given stroke is influenced by the two previous strokes.) This assumption makes practical sense in the context of handwriting. A bigram model would calculate the probability of the chain code c^d of a given character c as below.

$$p(c^d) = p(s_1^d|w_s) \cdot p(s_2^d|s_1^d) \cdot p(s_2^d|s_3^d) \dots \cdot p(w_e|s_n^d) \tag{3.8}$$

The probability of two strokes following each other is approximated as:

$$p(s_i^d|s_{i-1}^d) = \frac{c(s_{i-1}^d s_i^d)}{c(s_{i-1}^d)} \tag{3.9}$$

$c(s_{i-1}^d s_i^d)$ is the count of s_{i-1}^d and s_i^d appearing together, and $c(s_{i-1}^d)$ is the count of just s_{i-1}^d. Smoothing is done in case one of the above counts is zero. This applies some adjustments to the probabilities/counts so as to create non-zero probabilities.

Perplexity is a commonly used metric to evaluate N-gram models. Given an N-gram model (constructed with a training set), the perplexity of a test set in effect describes *uncertainty* or *confusion* in predicting the test set using information present in the training set. This can be interpreted as the *uncertainty* in predicting the implement movements of a character in our context and by extension used to measure the regularity of writing

in a script. A script set with repeated patterns can be expected to have low perplexity. For instance, if there is a large number of characters in the training set which have the sequence $N\ E$ in them, then a test character containing $N\ E$ will register relatively low perplexity. But a character without $N\ E$ will produce high perplexity, as there is a certain amount of *uncertainty* in the latter (because it is not very predictable from the training set).

Angle-Based Metrics Analyzing the different angles of strokes occurring in a character can help to characterize a particular scribal behavior. Monolithic metrics such as angle of inclination and angle of writing, suggested as paleographic differentiators, were all subjected to various interpretations. However, having a structured representation of a character facilitates retrieving several metrics related to angles, which can be used individually or in combination as necessary. Thus, we can objectively measure all possible angles that can be used as a collective metric to quantify stroke behavior.

We define a few important angle-based metrics that could be used. The major angle is the angle of the largest primitive stroke present in the character. The initial angle is defined with the initial stroke. The divergence angle, the angle between a character's first and last points, could also be considered as a metric. For multistroke characters, the angle of pen-drag can be an important measure. Angles between disjoint strokes can be plotted as a histogram to visualize the changes in writing behavior. Apart from angles between two strokes that are calculated using the corresponding tangents as mentioned in section 3.2.2, the other angles are measured with respect to the corresponding horizontal baselines of a character's bounding box.

Pen-Drag Distance The pen-drag distance is a metric with respect to multi-stroke behavior. This captures the hand movements between pen-strokes, which are an important part of multi-stroke production.

Cognitive Information

Writing a character is usually a top-down process. A character has to be memorized and then reproduced. Consequently, this requires elaborate trajectory planning. To quantify cognitive information, we need to find out the approximate information required to memorize and produce characters.

In this respect, we refer to Algorithmic Information Theory (AIT). Especially within AIT, the *Kolmogorov complexity* attempts to find the minimal description of a given sequence (Wallace & Dowe, 1999). We also mention a related work by Isokoski (2001), who measures the complexity of characters by studying the number of straight lines required to approximate a character; it was a very subjective measure, however. In a similar way, we attempt to find out the minimal representation of a character required to reproduce it. Theoretically, this would consist of points necessary to plan the trajectory of a character. In fact, these directly correspond to the *segmentation points* in a character, as they define the character's shape. Hence, the count of segmentation points is a good indicator of the cognitive information in the character. However, the proximity and distribution of the segmentation points may also affect the information contained in the character. For instance, if several points are very close to each other it might create additional confounding factors to the trajectory planning, which will increase the information content. But for now, we can ignore such intricate details. These need to be studied in detail in the future.

The Ramer-Douglas-Peucker (RDP) algorithm (Douglas & Peucker, 1973) also computes the minimum number of points required to approximate a given curve. This mostly agrees with the number of segmentation points in some cases, but in other cases, this might not be so. The issue with RDP is that a threshold for approximation needs to be provided, depending on which we may get a slight over-estimation of the points required.

Both the RDP and number of segmentation points can be considered as different metrics that correspond to the cognitive information present in a character and can be used as required.

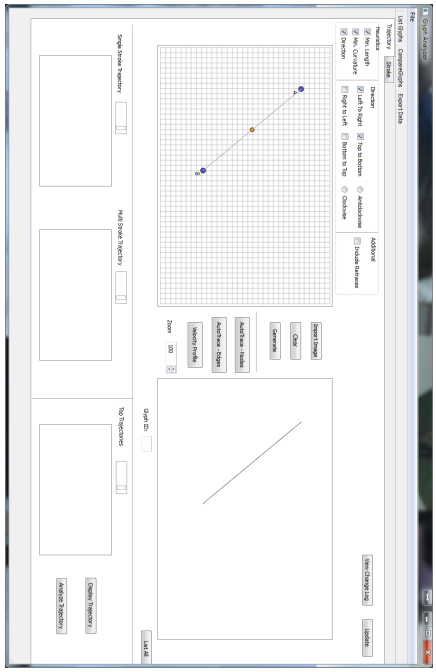
Metrics of Scripts vs. Metrics of Characters

Most of the metrics discussed were confined to those of individual characters. However, a script is a cohesive set of characters. In many cases, metrics of a script could be found just by averaging metrics of the individual characters. For instance, it would be possible to study the average curvature of various scripts and even compare them. Characters within a script are usually a heterogeneous set with different purposes and different patterns of usage. Hence, the average metric for a script may not always be appropriate. In such cases, instead of averaging the metrics, it is more useful to study the distribution of a metric in different scripts. If appropriate, a weighted average based on the frequency of usage could be considered. Since characters within a script behave as a set, studying the homogenization (or divergence) of properties within the script is a useful exercise. It would also be more useful if this could be overlaid with some other information such as usage frequency of characters. For instance, an interesting analysis would be to see how various properties of frequently used characters differ with respect to rarely used characters or in fact, if any such differences exist at all.

3.2.6 Prototype

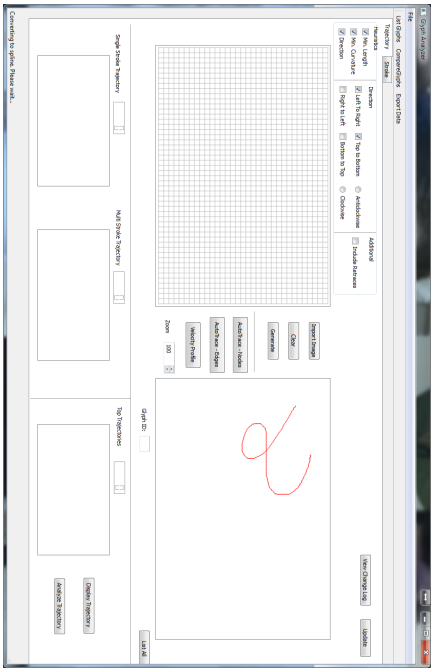
We have implemented a prototype of the proposed framework in Python 2.7. Python was chosen as it is simple and fast to prototype and can be easily extended as required using various libraries. The prototype developed implements all of the modules that were discussed in the previous sections. Users can start from importing a glyph image, then proceed to reconstruct the corresponding trajectory to retrieve the stroke structure and finally conclude with the extraction of metrics. Thus, the prototype allows users to seamlessly perform all the functionalities of the framework. Below we discuss in brief the related implementation details.

The prototype, as proposed in the framework, allows users to create spline representations in multiple ways as suited to them. They can explicitly define the B-spline segments manually if needed. This is done by first creating nodes and then creating the spline segments between the nodes by selecting the required node pairs. By default, the prototype



(a) B-spline for a straight line segment

(b) Manually created curve by manipulating the points of the B-spline from a straight line segment



(c) Hand-drawn glyph

(d) Spline representation of the drawn glyph

Figure 3.16: Spline representation

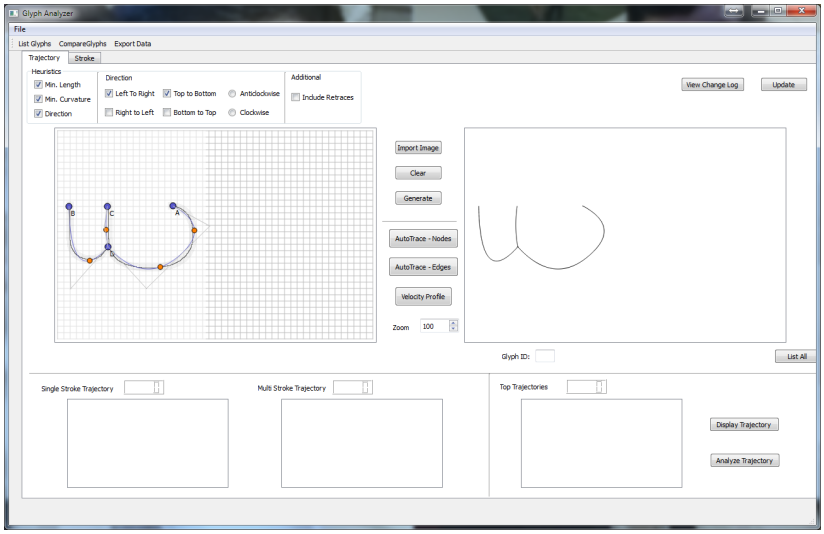
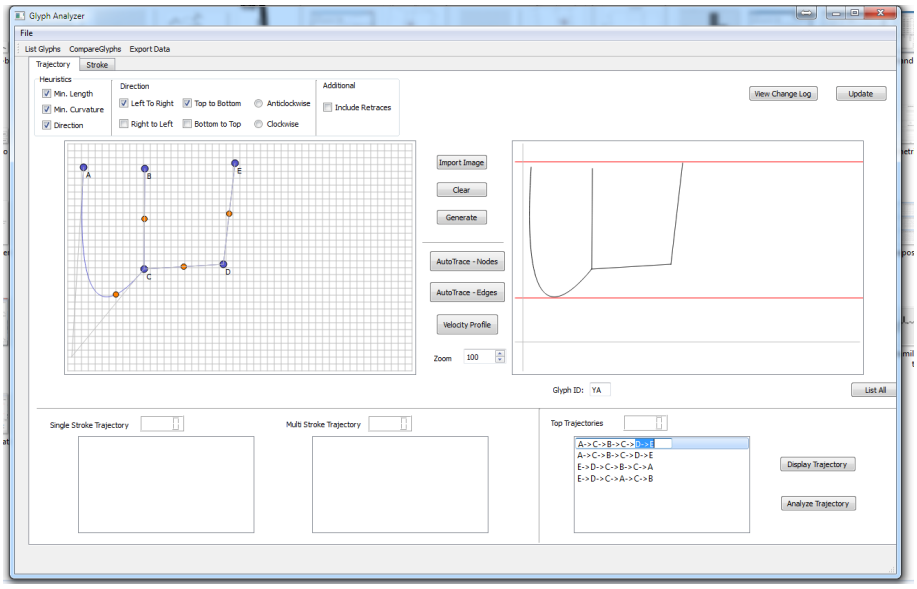


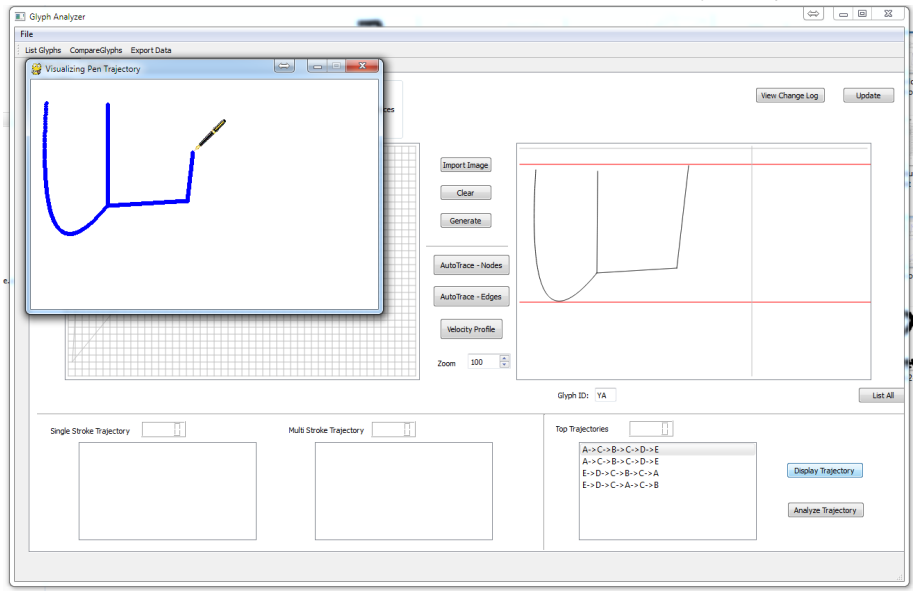
Figure 3.17: Imported image file with auto-traced nodes and edges and the corresponding spline representation

creates a straight line segment (see figures 3.16a & 3.16b), which can be modified to the desired shape by adjusting the points of the B-spline. To have a finer control over the shape, additional points can be created, deleted or moved, if needed. Users can also draw a character manually to create a spline representation. This is particularly useful (and in fact, faster) if they have access to graphic tablets like Wacom. The prototype has a custom module to deal with such input, and then reduces it into splines as seen in figures 3.16c and 3.16d. Finally, users are also able to import images and convert them into splines. The prototype auto-detects nodes and then proceeds to auto-trace edges as shown in figure 3.17. This process also allows manual intervention. Users can thus choose the level of automation that they find comfortable. The left-hand panel shows the spline representation, and the right-hand panel the resultant static shape of the spline representation as seen in figure 3.17. Simultaneously with the spline representation, the higher level graph abstraction is also created for that character internally.

Users can then proceed to reconstruct the corresponding trajectory by clicking the *Generate* button. The prototype suggests the top 5 trajectories for a character, which are typically ordered on the basis of their viability. The heuristics that generate these trajectories can be adjusted using



(a) Grantha letter YA with reconstructed trajectory



(b) Trajectory of Grantha letter YA being animated

Figure 3.18: Trajectory reconstruction

various options provided in the prototype. They can also choose to override the suggested trajectory by editing them as shown in figure 3.18a. These generated trajectories can be further visualized as an animation for ease of viewing as illustrated in figure 3.18b as well.

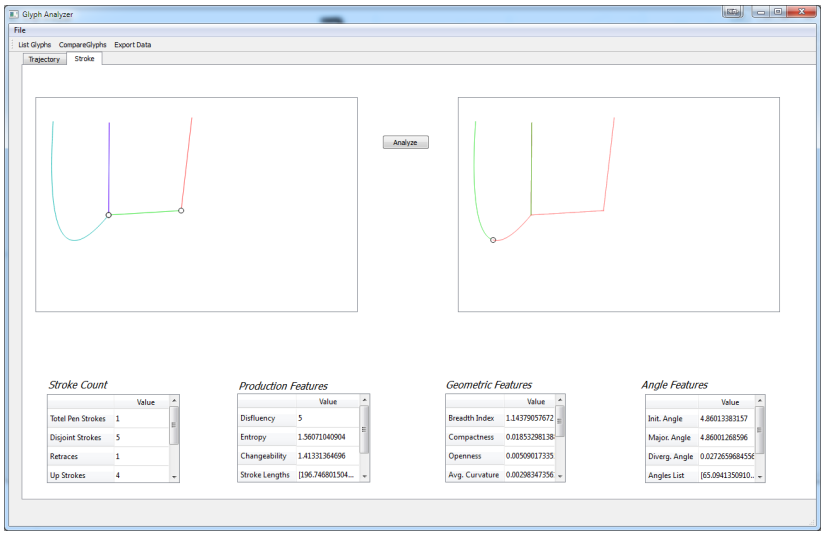


Figure 3.19: Grantha letter YA being stroke segmented

Users can now choose any of the five trajectories to be the basis of stroke segmentation. After selecting a particular trajectory, they can click the *Analyze Trajectory* button to view the segmentation results. The view now changes, with two different panes showing alternate abstractions of the segmentation process as seen in figure 3.19. The left pane shows a character segmented into disjoint strokes, while the right pane shows the primitive strokes. Existing points can be moved around and new segmentation points can be created as required. At this point, the prototype allows users to experiment with different trajectories, as each trajectory will yield a different kind of segmentation. Thus, the prototype allows users to explore different kinds of analysis. Based on the particular stroke segmentation, metrics are then extracted and displayed. As it is not particularly helpful to just display them, they can also be exported into CSV format for later quantitative analysis. Furthermore, all user involvements are logged and can be viewed when needed (see figure 3.20).

Once the analysis of a character is completed, it can now be saved

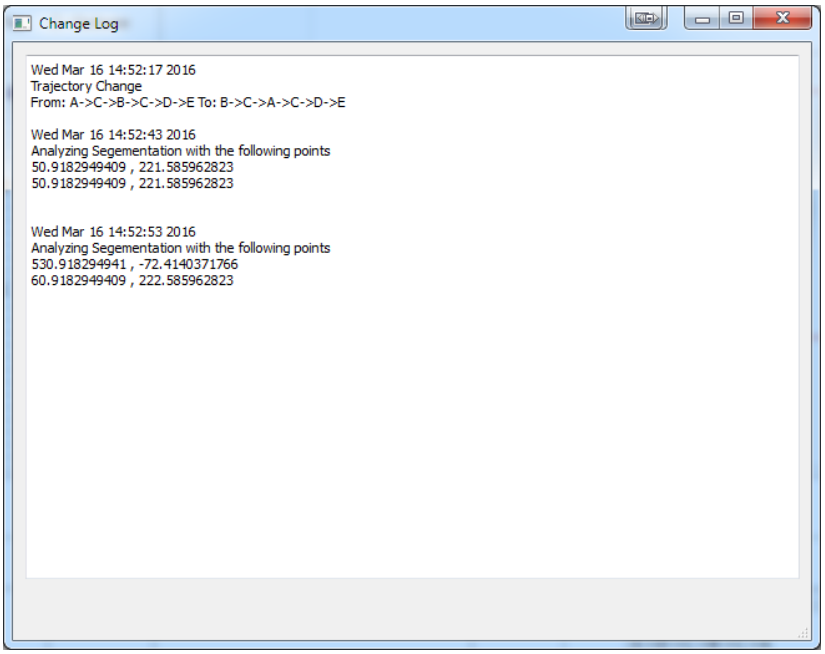


Figure 3.20: Log displaying list of changes made by users

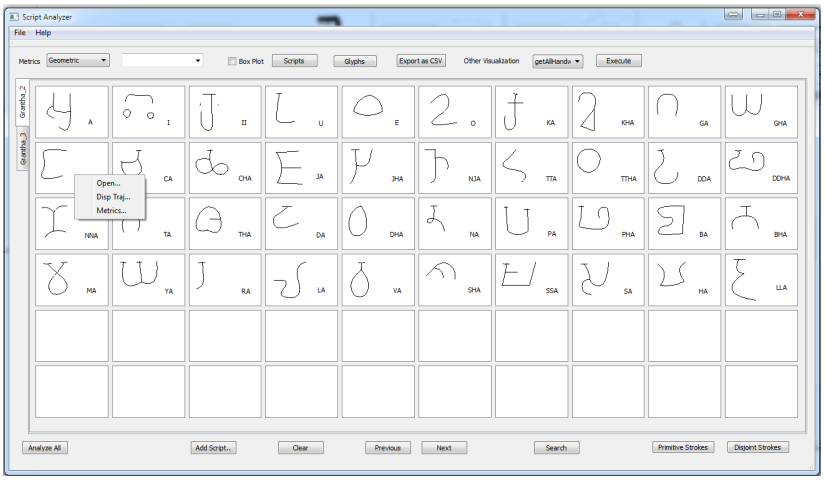


Figure 3.21: Screenshot of a script repository

for future use. Analyzed characters can be named and then saved under a script set, and can be viewed or modified when needed. This creates a script repository, which contains a collection of characters that have been analyzed and saved. Multiple scripts can also be loaded as shown in figure 3.21. Users also have the option to view the script inventory (derived from the stoke representation of the individual characters) of a given script.

Future Enhancements

From the perspective of paleographic scripts, character images are usually very noisy and importing them would require several layers of pre-processing and noise-removal. Image processing in the prototype as of now assumes the presence of clean images and hence adequate support for noisy images must be added. Also, many historical scripts have standardized font representations, which already have character glyphs as Bezier curves. Adding the option to import TrueType outlines from fonts would also be a very useful feature. Trajectory reconstruction has been implemented only for unistroke characters. Reconstruction for multistroke characters is more complicated and requires several layers of additional heuristics to order pen strokes. Hence, for multistroke characters the trajectory has to be manually entered. However, a primitive implementation does exist in the code base for them, which just needs to be made more rigorous.

Open Source Release

For the benefit of the wider research community, the prototype software has been released under the open source GNU GPL v3 license. Researchers can now access the prototype and use it for their quantitative and descriptive analysis. The prerequisites for running the prototype are Python and its associated libraries namely, *Pygame, Pyside, Numpy, Scipy, Pylab, Sympy, Matplotlib, CV2 and Pandas*. The code can be obtained from the following URL:

https://github.com/virtualvinodh/scriptanalyzer

3.3 Handwriting Modeling of Shape Changes

In section 1.1, we mentioned the fact that Stansbury (2009) highlighted how digital paleography should focus on exploring the ways strokes have evolved. Similarly, in the same section we discussed that Mallon (1952) and Blanchard (1999) performed descriptive paleography to analyze the evolution of the minuscule alphabets from the majuscule alphabets, in terms of changes in scribal behavior. Even though our metrics can quantify various aspects of scribal behavior, they do not directly quantify the changes themselves. However, with the aid of our analysis framework we can resort to handwriting modeling, which directly quantifies those changes.

Herzog et al. (2010) argue that handwriting models cannot be applied to paleography, claiming they are unsuitable for historical scripts, but they do not offer any proper rationale for their claims. There are some models that actually consider the fundamental motions of handwriting, which we believe can be extremely useful in paleographic analysis. Except for very specific writing implements/modes, at a high-level understanding, basic handwriting motion (see §3.1) overall remains the same. Even if the models are not considered to be completely applicable to a particular mode of historical writing, they can throw some light on the overall fundamental handwriting behavior, which can still be useful to computationally describe scribal behavior. Also, the underlying mechanism of these models can always be interpreted (or adjusted) in terms of a specific writing process.

3.3.1 Sigma-Lognormal Model of Handwriting Generation

In section 2.3, we elaborated on various kinds of handwriting models that are available and we also included a brief introduction to the Sigma-Lognormal model. We propose using this model to computationally describe the shape changes in the paleographic data. To briefly recollect, handwriting motion is assumed to be composed of multiple primitive strokes, which seamlessly integrate to form the handwriting motion.

Each of these primitive strokes have a bell-shaped velocity profile and can be modeled as a mathematical function of six parameters, with each parameter representing a different aspect of the strokes. Any reference to strokes in the below sections refers specifically to primitive strokes unless otherwise mentioned. A stroke s_i is given by:

$$s_i = f(D_i, t_0^{(i)}, \theta_s^{(i)}, \theta_e^{(i)}, \mu_i, \sigma_i) \tag{3.10}$$

D - amplitude of a stroke i.e. the length
t_0 - time of occurrence of a stroke i.e. the starting time
θ_s - starting angle of a stroke
θ_e - ending angle of a stroke
μ - time-delay of the neuromuscular system for a stroke
σ - response time of the neuromuscular system for a stroke

The stroke s_i is assumed to occur along a fixed pivot with a certain amplitude. It is constructed as an arc whose initial and final deviation from the pivot is given by the angle parameters. It can be seen that the model also has parameters that correspond to the neuromuscular system, apart from those that describe its physical attributes.

The model is further elaborated below with the relevant equations modeling a character's production velocity and curvature, which are then used to predict the positions of an implement on a surface i.e. the shape of a character.

The velocity profile v_i of s_i as a function of time is given by:

$$v_i(t) = \frac{D_i}{\sigma_i(t - t_0^{(i)})\sqrt{2\pi}} e^{-\frac{1}{2\sigma_i^2}(ln(t - t_0^{(i)}) - \mu_i)^2} \tag{3.11}$$

The corresponding curvature ϕ_i of s_i is given by:

$$\phi_i(t) = \theta_s^{(i)} + \frac{\theta_e^{(i)} - \theta_s^{(i)}}{2}\left(1 + erf\left(\frac{ln(t - t_0^{(i)}) - \mu_i}{\sigma_i}\right)\right) \tag{3.12}$$

where *erf* is the error function:

$$erf(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2} dt \tag{3.13}$$

The overall velocity profile v of a character in the x and y directions is given as the sum of the individual velocity components corresponding to each stroke. Thus, individual strokes have an impact on the overall velocity profile and therefore on the final shape of the character.

$$v_x(t) = \sum_{i=1}^{n} v_i(t) \cos \phi_i(t) \tag{3.14}$$

$$v_y(t) = \sum_{i=1}^{n} v_i(t) \sin \phi_i(t) \tag{3.15}$$

where n is the number of strokes in a character.

The actual position of a writing implement in the x and y directions at a given time t on a surface is calculated using the overall velocity profile as:

$$p_x(t) = p_x^0 + \int_0^t v_x(\tau) d\tau \tag{3.16}$$

$$p_y(t) = p_y^0 + \int_0^t v_y(\tau) d\tau \tag{3.17}$$

where p_x^0 and p_y^0 refer to the initial positions of the implement in the x and y directions.

$$p(t) = \begin{pmatrix} p_x(t) \\ p_y(t) \end{pmatrix} \tag{3.18}$$

where $p_x(t)$ and $p_x(t)$ are the positions in the x and y directions, and $p(t)$ is the overall position of an implement at a given time.

A given character c is a collection of all the positions of a writing implement on a given surface as given by equation 3.18. Thus, c can now be mathematically defined in terms of a parameter matrix m_c, where each row in the matrix corresponds to the parameter vector of the strokes that they model.

$$m_c = \begin{pmatrix} D_1 & t_0^{(1)} & \theta_s^{(1)} & \theta_e^{(1)} & \mu_1 & \sigma_1 \\ D_2 & t_0^{(2)} & \theta_s^{(2)} & \theta_e^{(2)} & \mu_2 & \sigma_2 \\ \dots & & & & & \\ D_n & t_0^{(n)} & \theta_s^{(n)} & \theta_e^{(n)} & \mu_n & \sigma_n \end{pmatrix} \tag{3.19}$$

$$c = f(m_c) \tag{3.20}$$

By modifying these six parameters for each of the strokes, the shape of a character can be manipulated as required. However, the model can be even further simplified to suit our purposes. μ and σ are peripheral features that do not fundamentally affect the shape of a character and only create slight changes in the shape. Ignoring these non-critical parameters (and considering them to be a constant), the model can now be approximated to just four parameters. Subsequently, these four parameters can be directly mapped to the scribal handwriting behavior that produced the character.

$$m_c \approx \begin{pmatrix} D_1 & t_0^{(1)} & \theta_s^{(1)} & \theta_e^{(1)} \\ D_2 & t_0^{(2)} & \theta_s^{(2)} & \theta_e^{(2)} \\ \dots & & & \\ D_n & t_0^{(n)} & \theta_s^{(n)} & \theta_e^{(n)} \end{pmatrix} \tag{3.21}$$

We can transform a character c_1 to c_2 (with parameter matrices m_c^1 and m_c^2) by means of a transformation matrix T, which modifies the Sigma-Lognormal parameters of the original character to bring about the shape changes that would lead to the formation of c_2.

$$T = \begin{pmatrix} \pm\delta_{11} & \pm\delta_{12} & \pm\delta_{13} & \pm\delta_{14} \\ \pm\delta_{21} & \pm\delta_{22} & \pm\delta_{23} & \pm\delta_{24} \\ \dots & & & \\ \pm\delta_{n1} & \pm\delta_{n2} & \pm\delta_{n3} & \pm\delta_{n4} \end{pmatrix} \tag{3.22}$$

where δ_{ij} is a change to an individual parameter.

$$c_2 \approx f(m_c^1 + T) \tag{3.23}$$

$$m_c^2 = m_c^1 + T \tag{3.24}$$

$$m_c^2 = \begin{pmatrix} D_1 \pm \delta_{11} & t_0^{(1)} \pm \delta_{12} & \theta_s^{(1)} \pm \delta_{13} & \theta_e^{(1)} \pm \delta_{14} \\ D_2 \pm \delta_{21} & t_0^{(2)} \pm \delta_{22} & \theta_s^{(2)} \pm \delta_{23} & \theta_e^{(2)} \pm \delta_{24} \\ \dots & & & \\ D_n \pm \delta_{n1} & t_0^{(n)} \pm \delta_{n2} & \theta_s^{(n)} \pm \delta_{n3} & \theta_e^{(n)} \pm \delta_{n4} \end{pmatrix} \tag{3.25}$$

The transformation matrix T thus quantifies all the changes in the writing behavior in terms of the Sigma-Lognormal parameters.

3.3.2 Modeling Stroke Phenomena

As briefly mentioned in section 1.1, Blanchard (1999) notes that strokes exhibit various interesting behaviors, which he calls *Graphétique* (Graphetics, similar to Phonetics). He notices common phenomena such as *rounding of corners/angles* and *ligations between strokes*, and also underlines that they are very important to the study of fundamental paleographic phenomena. He uses several complex writing behaviors to describe the evolution of lowercase Greek letters from the uppercase letters. These can broadly be described using a combination of the following basic stroke phenomena and the Sigma-Lognormal model allows modeling all of them computationally.

1. Change in length
2. Change in curvature
3. Ligation of strokes

The simplest phenomenon of them all is the change in stroke length. It can be straightforwardly modified by changing the magnitude of parameter D.

The difference in the angle parameters of a stroke can be taken as a single aggregate entity that controls the curvature of a stroke.

$$\Delta\theta_i = \theta_e^{(i)} - \theta_s^{(i)} \tag{3.26}$$

If $\Delta\theta_i$ tends towards zero, the stroke will be a straight line; as $\Delta\theta_i$ moves towards 2π radians it becomes increasingly circular. If the difference is positive, the stroke is convexly curved otherwise it is concave.

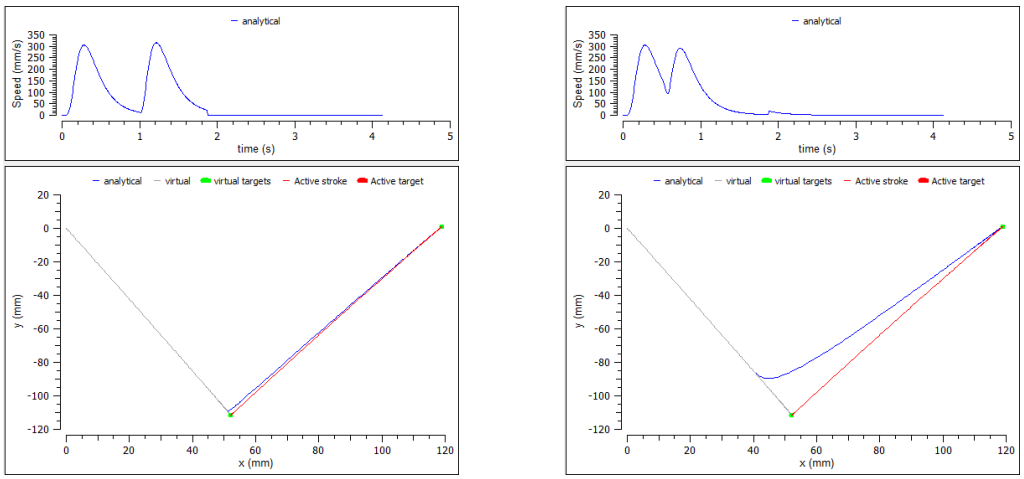
As seen earlier, t_0 represents the time when stroke production begins. The interaction between two strokes can be altered by changing the difference between their stroke initiation times.

$$\Delta t_0^{(i);(i+1)} = t_0^{(i+1)} - t_0^{(i)} \tag{3.27}$$

In the Sigma-Lognormal model, the overall trajectory of a character is defined using an action plan made up of a set of virtual targets. These are imaginary way-points that define the evolution of the trajectory over time. When a stroke aimed towards the next target begins earlier than the current active stroke, it results in the overlap of the two velocity profiles forming a smoother trajectory and thus causing ligation (Berio & Leymarie, 2015). This can be achieved by decreasing $\Delta t_0^{(i);(i+1)}$, which causes the merger of the adjacent strokes s_i and s_{i+1}. Essentially, this would make two disjoint strokes having a sharp corner to form a rounded corner. Figures 3.22a and 3.22b show ligation performed using the Sigma-Lognormal model. Particularly note that the velocity profile in the figure 3.22b shows a greater degree of overlap compared to figure 3.22a, which is correspondingly reflected in the shape. On the other hand, if $\Delta t_0^{(i);(i+1)}$ is increased, it causes two velocity profiles to separate further and creates a more discontinuous trajectory.

By changing the four parameters D, t_0, θ_s and θ_e, the overall shape of a character can be manipulated by concurrent applications of the above transformations to the individual strokes. Thus, the Sigma-Lognormal model enables us to model and execute changes in complex writing behavior.

While the Sigma-Lognormal model can be used to computationally describe and model various phenomena affecting the shape of the strokes contained in a character, it is not possible to model stroke augmentations. In some cases, additional strokes are augmented (intentionally or unintentionally) into a character. Whilst this phenomenon is important, it cannot be directly modeled. However, the model can execute the re-



(a) Disjoint strokes with a sharp corner

(b) Disjoint strokes with rounded corners through ligation

Figure 3.22: Stroke ligation. The green points are the virtual targets for the strokes.

duction of a stroke by nullifying the parameters of the corresponding stroke. This can be used to create a possible work-around. We could, in principle, reverse the development for the sake of analysis. Consider two characters c_1 and c_2 (with parameter matrices m_c^1 and m_c^2), where c_2 has more strokes than c_1. The Sigma-Lognormal model can be used to computationally describe the shape change of c_2 to c_1, which can give us the corresponding transformation matrix T. The transformation of c_1 to c_2 can then be modeled in a straightforward way as below.

$$c_1 \approx f(m_c^2 + T) \tag{3.28}$$

$$m_c^1 = m_c^2 + T \tag{3.29}$$

$$m_c^1 - T = m_c^2 \tag{3.30}$$

$$m_c^1 + (-T) = m_c^2 \tag{3.31}$$

(-T) can be taken as the transformation matrix that changes c_1 to c_2.

3.3.3 Creating Sigma-Lognormal Models of Characters

Creation of Sigma-Lognormal models of characters requires their actual kinematic information (i.e. velocity profile). Usually, this would involve writing them through an appropriate digital interface (for instance, a drawing pad) and capturing their real-time velocity data, which is then used to fit corresponding Sigma-Lognormal models. However, capturing this real-time velocity information for each character in a large dataset is a very tedious task. Therefore, we attempt to synthesize the handwriting signals directly from the characters themselves.

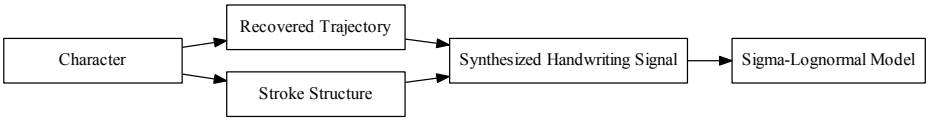


Figure 3.23: Synthesizing handwriting signals

It can be recalled that characters analyzed through our framework are pre-segmented (see §3.2.3) and their stroke structures (see §3.2.4) are already computed. Each of the primitive strokes in their structure approximately corresponds to a stroke of the Sigma-Lognormal model. In the case of multi-stroke characters, the invisible pen-drag is modeled as an additional visible stroke (with respect to the handwriting model) as well. This precomputed structure along with the recovered trajectory information (which provides the direction and ordering for the strokes) can be used for handwriting modeling without any further processing. We only have to reconstruct an ideal bell-shaped velocity profile that follows a log-normal distribution for the constituent strokes of a character as demanded by the model. This can be done through the straightforward application of statistical methods.

The log-normal distribution for a variable x is given by the following function:

$$P(x) = \frac{1}{\sigma\sqrt{2\pi}x} e^{\frac{-(ln(x)-\mu)^2}{2\sigma^2}} \tag{3.32}$$

where σ is the standard deviation and μ is the mean of the distribution.

Rewriting the equation in terms of distance and time to calculate the velocity of writing we get:

$$v(t) = \frac{d}{t\sigma\sqrt{2\pi}} e^{\frac{-(ln(t)-\mu)^2}{2\sigma^2}} \tag{3.33}$$

Where d is the length of a stroke and t is the time taken to write it.

With reasonable assumptions for the standard deviation, mean and also the average speed of writing, we can recreate the ideal velocity profile for a character. This synthesized signal can then be used for the creation of a handwriting model for the character. But do note that this is a very idealized reconstruction. Accordingly, the parameters derived from them are also very much idealized. However, they do serve as a good approximation for the real handwriting behavior.

In collaboration with École Polytechnique de Montréal, we were provided with the *Script Studio* collection, which encompasses different modules that implement various functionalities of the Sigma-Lognormal model and provides an integrated environment to construct and experiment with the models of characters. Script Studio consists of two main modules. The first module termed *SimScript* (O'Reilly & Plamondon, 2007, 2009b) allows users to generate 2D trajectories of handwriting using Sigma-Lognormal equations for handwriting modeling. The second module called *XzeroROBUSTE* (O'Reilly & Plamondon, 2009a; Djioua & Plamondon, 2009) is a system that allows the automatic extraction of the Sigma-Lognormal parameters from handwritten specimens. We would be using both these modules to implement handwriting modeling. Though the software is not open source, it is available for research purposes upon request from the corresponding researchers.

It is possible to fit Sigma-Lognormal parameters for any character using Script Studio, given the corresponding handwriting signal. The number of Sigma-Lognormal parameter vectors as derived from the software usually corresponds to the actual stroke count of a character. However, in some cases the tool might hypothesize additional underlying strokes to model the visual strokes in the character. It can be recalled (see §3.2.3) that we assume that visible strokes are good approximations of the

underlying strokes and use this to segment characters. Such additional strokes (posited by the model) must be taken into special consideration, while using the model to describe scribal behavior.

3.3.4 Modeling of Shape Changes

Figure 3.24: Development of early Tamil *O* [Right] from Brahmi *O* [Left]

We illustrate the use of the Sigma-Lognormal model by mathematically modeling the shape change of a character. Figure 3.24 shows the development of early Tamil character *O* from its Brahmi script ancestor. The trajectories of the characters are also shown in the same figure.

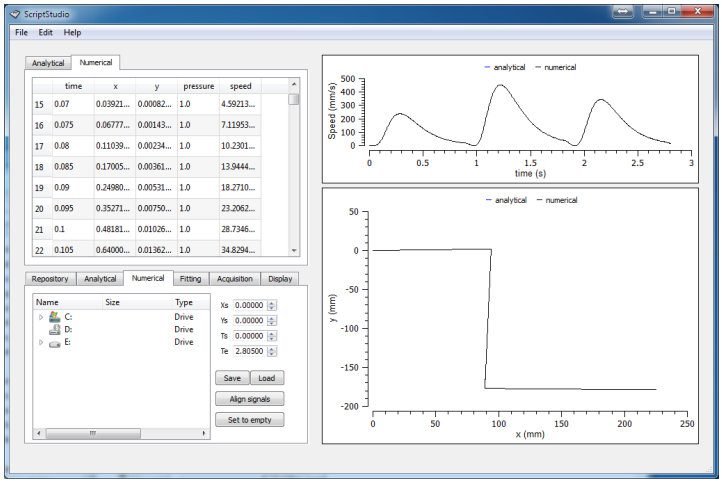


Figure 3.25: Sythesized handwriting signal for Brahmi *O*

The Brahmi character was initially analyzed through our framework prototype and decomposed into strokes. The handwriting signal for it was also then synthesized using the framework prototype that we developed. Figure 3.25 shows the synthesized handwriting signal within the Script Studio tool. We can clearly see three bell-shaped velocity curves corresponding to the three strokes that make up the character. We then

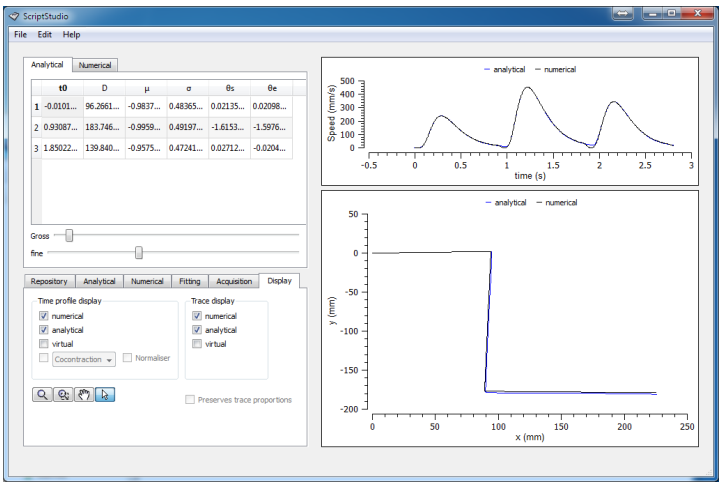


Figure 3.26: Fitted Sigma-Lognormal parameters for the reconstructed velocity profile of Brahmi O

used the tool to extract the Sigma-Lognormal parameters corresponding to the three strokes as seen in figure 3.26. Thus, we have constructed a computational model that describes the generation of the character, which is now represented by a collection of three vectors.

From figure 3.24, we can clearly hypothesize a number of stroke phenomena that could have caused the shape change such as the ligation of strokes and curving of the straight strokes. As seen in section 3.3.2, the Sigma-Lognormal model performs all these operations.

The character c_O, corresponding to Brahmi O, is represented using the following parameter matrix m_c^O with each row vector corresponding to the three strokes of the character - s_1, s_2 and s_3.

$$m_c^O = \begin{pmatrix} 96.2661 & -0.0101 & 0.02135 & 0.02098 \\ 183.746 & 0.93087 & -1.6153 & -1.5976 \\ 139.840 & 1.85022 & 0.02712 & -0.0204 \end{pmatrix} \tag{3.34}$$

We will begin by modeling the ligation of strokes. The current time between adjacent strokes are as follows: $\Delta t_0^{(1);(2)} = 0.94097$ and $\Delta t_0^{(2);(3)} = 0.91935$. By reducing $t_0^{(2)}$ and $t_0^{(3)}$, we can reduce the corresponding time difference. To effect this, $\Delta t_0^{(1);(2)}$ and $\Delta t_0^{(2);(3)}$ are reduced by 60% by manipulating s_2 and s_3 to be produced very early. This results in the following matrix with the resulting shape change as shown in figure 3.27.

$$m_c^O = \begin{pmatrix} 96.2661 & -0.0101 & 0.02135 & 0.02098 \\ 183.746 & 0.33087 & -1.6153 & -1.5976 \\ 139.840 & 0.65022 & 0.02712 & -0.0204 \end{pmatrix} \tag{3.35}$$

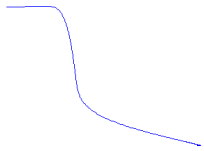


Figure 3.27: Shape change after manipulation of stroke timings

Changes in $\Delta t_0^{(i);(i+1)}$ could be manipulated by changing the stroke initiation times of s_i, s_{i+1} or both. Mathematically, we are just interested in the overall increase or decrease in $\Delta t_0^{(i);(i+1)}$. However, this manipulation has to be pragmatically decided. For instance, in our case the decrease in $\Delta t_0^{(1);(2)}$ could have been performed by either delaying s_1 (i.e. increase $t_0^{(1)}$) or advancing s_2 (i.e. decrease $t_0^{(2)}$). s_1 being the first stroke of the character, it does not make any real-world sense to delay its initiation. It is more pragmatic to consider s_2 was initiated early (probably as a result of writing fast). In the same way, for decreasing $\Delta t_0^{(2);(3)}$, we advance s_3 since s_2 has been already initiated earlier (and hence cannot be delayed).

The next step would be to change the curvature of strokes. We can see that in Brahmi the strokes are nearly straight lines given by their $\Delta\theta$ values: $\Delta\theta_1 \approx 0$, $\Delta\theta_2 \approx 0$, $\Delta\theta_3 \approx 0$. By changing the stroke angle values for all the three strokes to make them more curvy, we get the below matrix. They now have a proper curvature as shown in figure 3.28. In fact, the $\Delta\theta_1$ for s_1 is nearly 5 rad i.e. 280°.

$$m_c^O = \begin{pmatrix} 96.2661 & -0.0101 & 1.4608 & -3.6592 \\ 183.746 & 0.33087 & 4.6678 & 3.8848 \\ 139.840 & 0.65022 & 0.1064 & -0.3008 \end{pmatrix} \tag{3.36}$$

Similar to Δt_0, $\Delta\theta$ can be adjusted either by manipulating θ_s, θ_e or even both. This is decided pragmatically as well. For instance, in some cases, changing only θ_e or θ_s could place a stroke in an inappropriate angle. Hence, it may be necessary to adjust them both simultaneously, as was done in our case.

Figure 3.28: Shape change after manipulation of stroke curvatures

From figure 3.28, we can see that the character has still not reached the shape of early Tamil *O*. This is because the apparent length of the stroke *s* became reduced due to the great increase in its curvature. Therefore, the magnitude of the stroke needs to be increased as below to finally reach the target shape as seen in figure 3.29.

$$m_c^O = \begin{pmatrix} 194.2661 & -0.0101 & 1.4608 & -3.6792 \\ 183.746 & 0.33087 & 4.6678 & 3.8848 \\ 139.840 & 0.65022 & 0.1064 & -0.3008 \end{pmatrix} \tag{3.37}$$

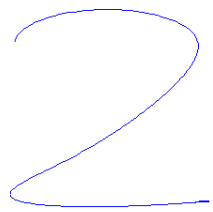


Figure 3.29: Shape change after manipulation of stroke lengths

The transformation of the initial shape to the target shape can finally be captured as a transformation matrix as shown below, with each row vector corresponding to the transformation of a stroke. Figure 3.30 visually summarizes the transformation.

$$T_O = \begin{pmatrix} 98 & 0 & 1.4395 & -3.6802 \\ 0 & -0.6 & 6.281 & 5.48248 \\ 0 & -1.2 & 0.0793 & 0.3212 \end{pmatrix} \tag{3.38}$$

The transformation matrix 3.38 quantifies all scribal behavior changes that caused the shape to change. For instance, it can be distinctly seen from the negative value for δt_{0_2} and δt_{0_3} that these strokes started early. Similar inferences can be made for curvatures and length as well. But

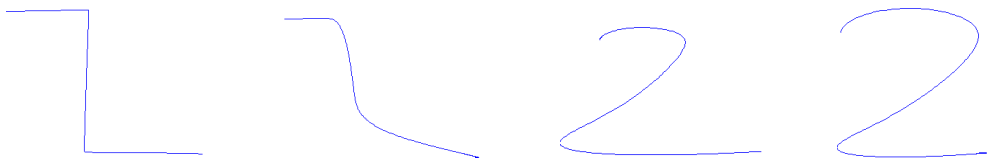


Figure 3.30: Different stages of handwriting modeling of character *O*

do note that the stages in figure 3.30 do not represent intermediate forms. They are just the resulting transformations of applying each stroke phenomenon. They could have been made in any order. The aim here is to get a final transformation matrix that would describe the changes in scribal behavior computationally.

However, the model can also be used to propose putative intermediate forms by applying the same stroke phenomena. This requires properly hypothesizing the underlying stroke phenomena for the intermediate forms. Figure 3.31 proposes proper intermediate forms for the transformation discussed earlier. We initially assume that scribes intend to change the angle of s_2 making it increasingly diagonal (top right). Then, perhaps they smoothened the corners, enabling them to write the character faster (bottom left). This probably resulted in further curving s_1 along with a corresponding increase in its length, thus giving us the final form of the character. This is one of the ways the character could have developed. It is quite possible to propose other intermediate forms based on different assumptions.

We can clearly see that using handwriting modeling, it is possible to quantify scribal behavioral change. It can be used to understand and model various phenomena of that behavior, which facilitates looking for patterns in such changes. Through this, we can obtain an overall view of the handwriting process and also the interaction of various factors, which can help us to understand the shape changing process. It can additionally be used to experiment with various writing behaviors by hypothesizing different underlying stroke behavior. This is particularly useful to construct putative intermediate forms as demonstrated earlier, and generate scribal variants of characters.

However, the quantification should not be taken as absolute. It must

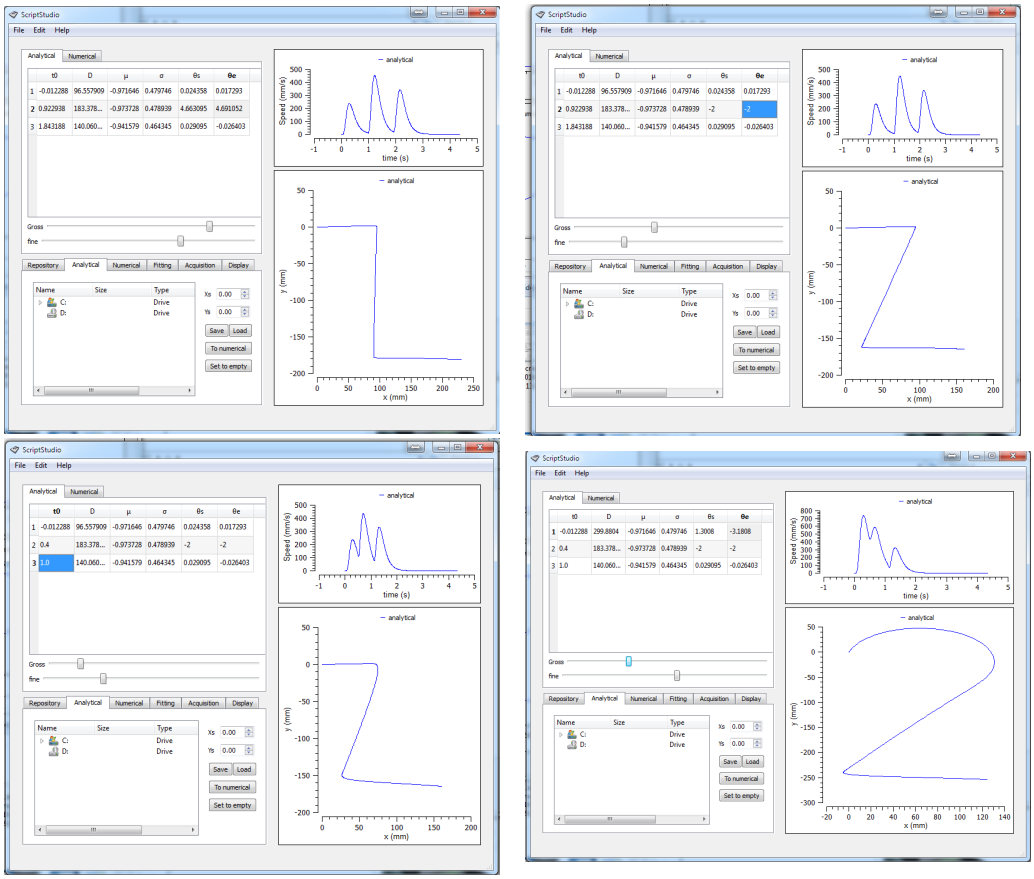


Figure 3.31: Constructing putative intermediate forms for Brahmi *O*

be noted that even slight changes in the shape of a character will result in two different Sigma-Lognormal values. Some parameters such as θ_s and θ_e are very sensitive. Also, in some cases, it is not possible to attain the target shape exactly and transformations are only approximate. The order of magnitude and the sign of the parameters (rather than specific values) should be taken as a generic indicators of the change in writing behavior. Thus, handwriting modeling is not entirely perfect, but it can model the overall stroke phenomena to a sufficient extent and provide a generic view on the scribal writing behavior.

3.4 Summary

We described a computational framework that analyzes characters using recovered handwriting information in the context of quantitative paleo-

graphy. It consists of five modules that process a character from its raw image-based representation into a natural stroke-based representation to extract various metrics. The functionality of these modules along with their underlying assumptions was detailed. We then proceeded to define metrics that can quantify handwriting information under three different categories - visual, kinematic and cognitive. Further, the use of the SigmaLognormal model of handwriting in conjunction with our framework was elaborated and it was shown to have applications in analyzing shape changes of characters and thus in understanding scribal handwriting behavior. In the next chapter, we will discuss a case study, which applies the methods and metrics presented in this chapter to study the development of Indic scripts.

Chapter 4

Case Study: Analysis of the Development of Indic Scripts

> अकारो मुखः सर्वधर्माणाम् आद्यनुत्पन्नत्वात्॥
>
> *akāro mukhaḥ sarvadharmāṇam ādyanutpannatvāt.*
>
> The letter 'A' is the door to all phenomena, for they are all originally unarisen.
>
> The Buddha[1]

India is linguistically a very diverse place, being the home of hundreds of languages spread across a billion people. This is also reflected in terms of diversity in scripts, as most of the major languages in the sub-continent have their own scripts that are as diverse as the languages themselves. But it may be a surprise to find out that many of the Indic scripts that exist today were all derived from the same source script *Brāhmi* (see figure 4.1). There have been several competing theories about the origins of Brahmi itself, but the general consensus is that it was largely inspired by or derived from the Aramaic script (Salomon, 1998). Perhaps due to its partially constructed nature, the initial shape of Brahmi was largely geometrical, but over time it has given rise to a wide variety of scripts, primarily due to variations in scribal handwriting.

[1] From The Perfection of Wisdom in 25000 verses (*Pañviṃśatisāhasrikā Prajñāpāramitā Sūtra*)

Indic scripts are among the few script families around the world that have existed more or less as a continuum for several centuries. For a major Indic script, we can derive an *almost* linear evolutionary line from Brahmi, which provides a unique opportunity to investigate the evolution of characters in terms of scribal behavior.

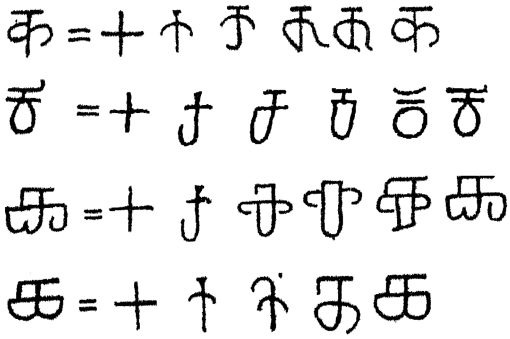


Figure 4.1: Development for Brahmi *KA* into *Devanagari*, *Kannada*, *Grantha* and *Tamil*

It may be recalled that in section 1.1 we mention Mallon et al. (1939) and Blanchard (1999) studying the development of minuscule letters from majuscule letters for Latin and Greek respectively. Along similar lines, we aim to study the development of Indic scripts using our proposed framework. We analyze the entire development of four major Indic scripts to demonstrate how our framework lends itself to the analysis of scribal behavior in a straightforward manner, and thus facilitates performing innovative and interesting quantitative analyses that describe the development. We present below a wide range of analyses that we perform on the script development. This helps us to uncover the complex interplay of handwriting factors which can identify salient features in handwriting and describe the changes in scribal behavior.

4.1 Data Set

To obtain a comprehensive view of the development, we have taken four major scripts belonging to the Brahmic family - *Devanagari*, *Tamil*, *Kannada* and *Grantha*, which we consider to be representative of the entire

script family. Ojha (1918) presents the development of these scripts in six distinct stages (with Brahmi being the initial stage), which we use for our case study. The detailed developmental stages of the scripts can be seen in figures 4.2, 4.3, 4.4 and 4.5. The progression of the scripts from one stage to another can be considered to have taken place in about 400 years and the entire development covers a period of approximately two millennia. It is to be noted that the scripts themselves show large geographical and scribal variations even over the same time period and only the normalized shapes are presented by Ojha. He manually performed the normalization by constructing shapes that attempted to capture the salient features of variant characters. As such the normalization tries to capture the generic scribal behavior for a given script in a specific time period rather than various individual scribal behaviors. This must be taken into consideration in our analyses. Also, he only considers characters that have consistent development from Brahmi and ignores those that had alternate developments (those that developed as variations of other base characters). In the previous figures, it can be noticed that some characters have fewer distinct stages compared to others. We consider those characters to have stablized early in the development. Therefore, we normalize the number of characters in the subsequent stages by retaining these stabilized characters in their character set. For instance, Tamil *PA* only shows three stages of development. Hence, the final stage of *PA* is also included in the character set of the later stages of Tamil to retain a constant character count. Grantha, Devanagari and Kannada have approximately 40 characters each in their repertoire, while Tamil has 20 characters. In total, with Brahmi as the common source script, we have 20 distinct stages of development consisting of approximately 730 unique characters in our dataset. Henceforth, the individual stages of the scripts are labeled from 1 to 6 with 1 referring to the original Brahmi stage.

Before proceeding with our analyses, we state the assumptions regarding our dataset. The normalized shapes of the characters as derived by Ojha (1918) are mainly based on epigraphic sources. They are not based on manuscript sources in which the bulk of the writing actually took place. However, there are only minor variations between the two, which makes it comparatively irrelevant when abstracting the development over

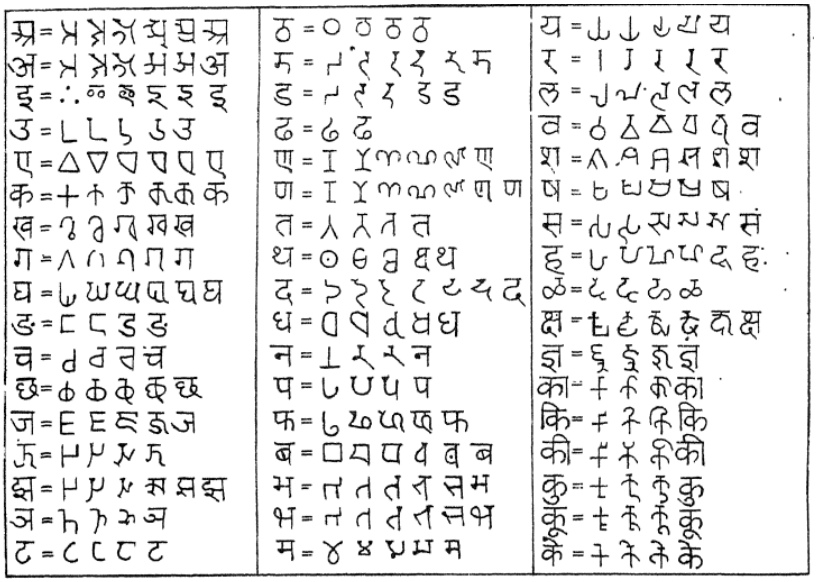


Figure 4.2: Development of Devanagari (Ojha, 1918, Plate LXXXIII)

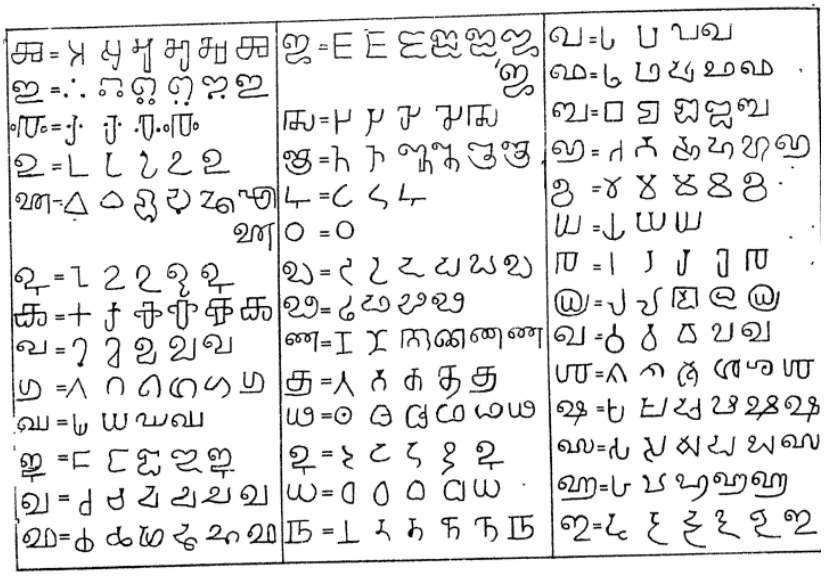


Figure 4.3: Development of Grantha (Ojha, 1918, Plate LXXXIV)

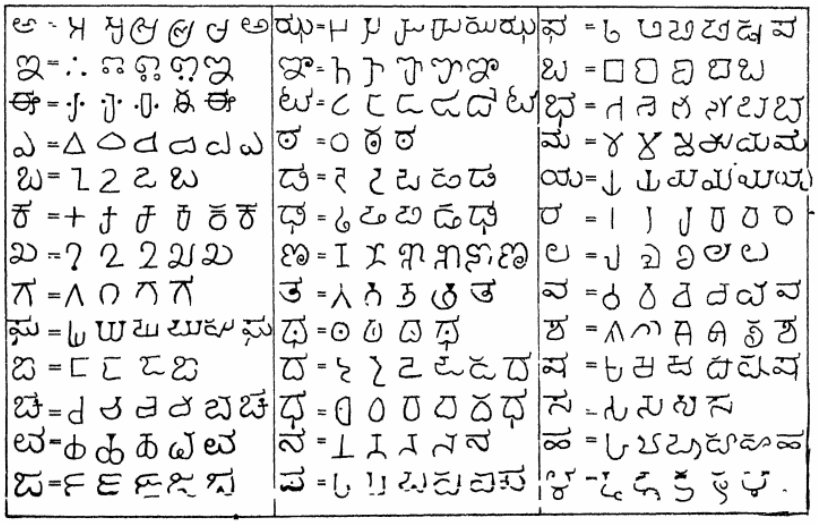


Figure 4.4: Development of Kannada (Ojha, 1918, Plate LXXXIII)

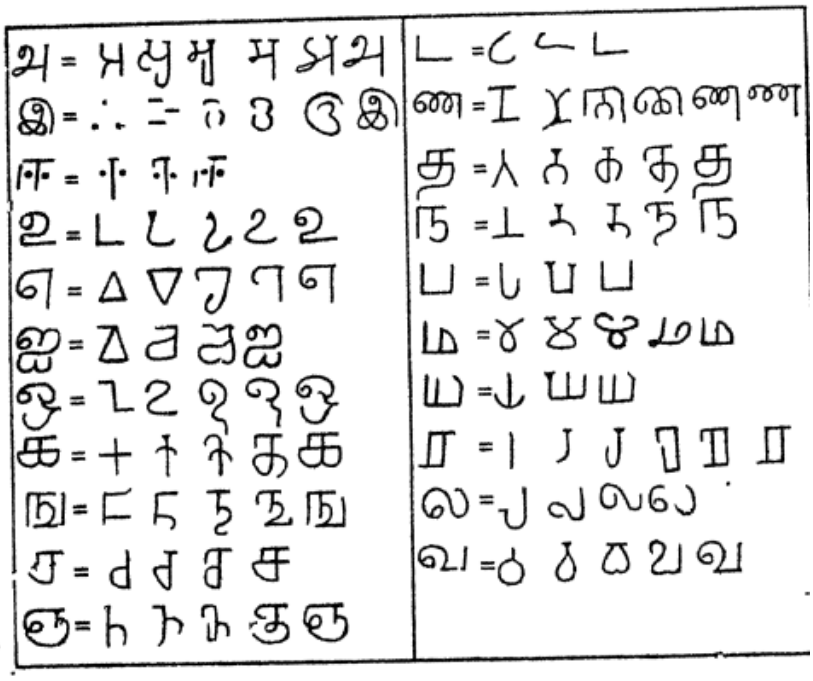


Figure 4.5: Development of Tamil (Ojha, 1918, Plate LXXXIV)

two millennia. Also, the different stages of the scripts are assumed to be temporally equidistant to each other for all practical purposes, and each stage in a script corresponds more-or-less directly to the corresponding stage in others, e.g. stage 2 of the four scripts is assumed to have occurred at approximately the same time. Also, while we discuss the changes in handwriting behavior, one important factor that must be considered is the writing implement and surface. For South Indian scripts (Tamil, Grantha and Kannada), writing was usually performed on palm leaves using an iron stylus(Kumar et al., 2009). The stylus was used to carve characters on dried palm leaves that were bound together. For Devanagari, writing was primarily performed with a reed pen (using ink) on tree-barks in addition to palm leaves(Lydon, 2015). It may be safely assumed that the implement and surface stayed consistent during the development until pre-modern times. While we present our results mostly independent of the implements and materials, we do speculate about their impact when necessary. The results and our subsequent interpretations must be read in the light of all these assumptions.

All of the characters in the script development are digitized and processed using the prototype of our analysis framework. Characters are first converted into splines, followed by reconstruction of their trajectories and then finally decomposed into their respective strokes (see §3.2). The trajectories and the corresponding segmentation results are checked (and corrected manually, if necessary) and then saved in a script repository corresponding to each script. In the end, we have the stroke structure of characters digitized and ready for metrics extraction. This pre-analyzed dataset is used for our further analyses. Figure 4.6 shows script repositories for the developmental stages of Grantha.

4.2 Quantitative Analysis of Metrics

Section 3.2.5 in the previous chapter proposes a number of useful metrics. Analyzing the development of the Indic scripts in the light of the metrics through quantitative methods provides an excellent opportunity to illustrate their use. From our digitized script repository, we extract nine major

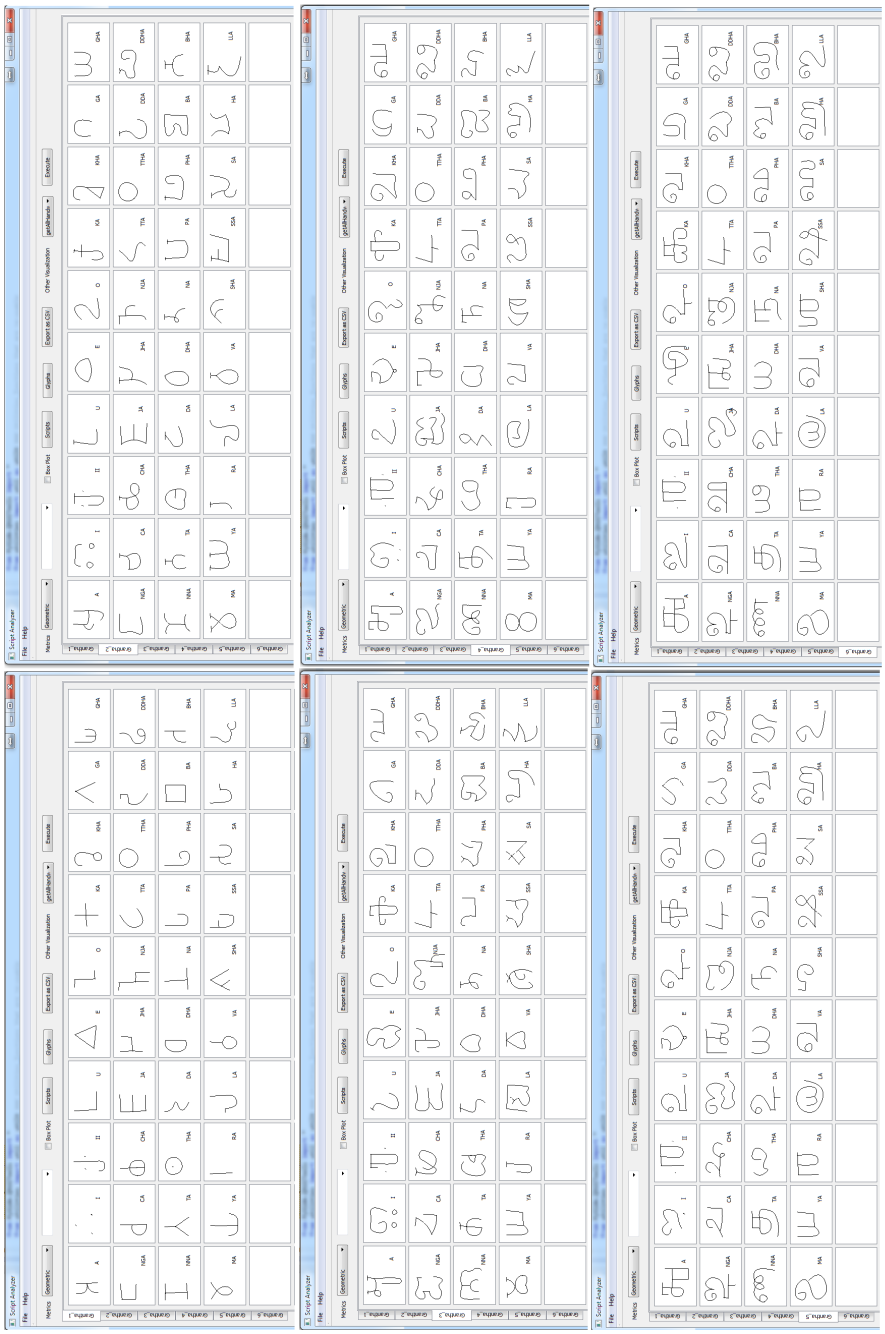


Figure 4.6: Script repositories for the development of Grantha

visual metrics: *Length, Size, Length-Breadth Index, Circularity, Rectangularity, Divergence, Openness, Average Curvature and Compactness* and twelve major kinematic metrics: *Pen-stroke Count, Disjoint Stroke Count, Retrace Count, Disfluency, Up-stroke Length, Down-stroke Length, Changeability, Entropy, Disjoint Angles Sum, Disjoint Angles Mean, Primitive Stroke Lengths Mean, Disjoint Stroke Lengths Mean*, which are used for subsequent analyses. We discuss below the various methods that we perform on the dataset to analyze overall behavior and stroke-level behavior, and further elaborate and interpret the results obtained. Whilst it is possible to perform a wide range of analyses, we perform only those that would simultaneously aid to understand the development and demonstrate the metrics' usefulness.

4.2.1 Trends in Handwriting

Evaluating changes in metrics over a period of time shows the general trends of various behaviors for each script, which can help to understand changes in writing. We analyze both visual and kinematic metrics for distinct patterns and trends.

Figure 4.7 shows the general trend in the averages of various visual metrics of scripts across the timescale of development. We can see that the size and length of characters steadily increase over time. Also, the length-breadth index indicates that characters are becoming wider. The outline of characters is approaching an ideal geometric shape as noted by the increase in circularity and rectangularity. This may be ascribed to the latent human nature to idealize the overall character outlines into symmetric shapes. In terms of pen positions, divergence is increasing over time, which appears to be a consequence of a corresponding increase in the length of characters. It takes more effort to maintain the starting and ending positions of an implement near each other. With respective to the total length, however, the pen positions become closer as shown by the decrease in openness. Compactness also appears to have dropped significantly. Brahmi had more strokes constricted into the same area with scribes incrementally spreading out the strokes later on. In terms of curvature, the latent trend is towards highly curved characters. This is understandable, as it has been suggested that it is easier for humans

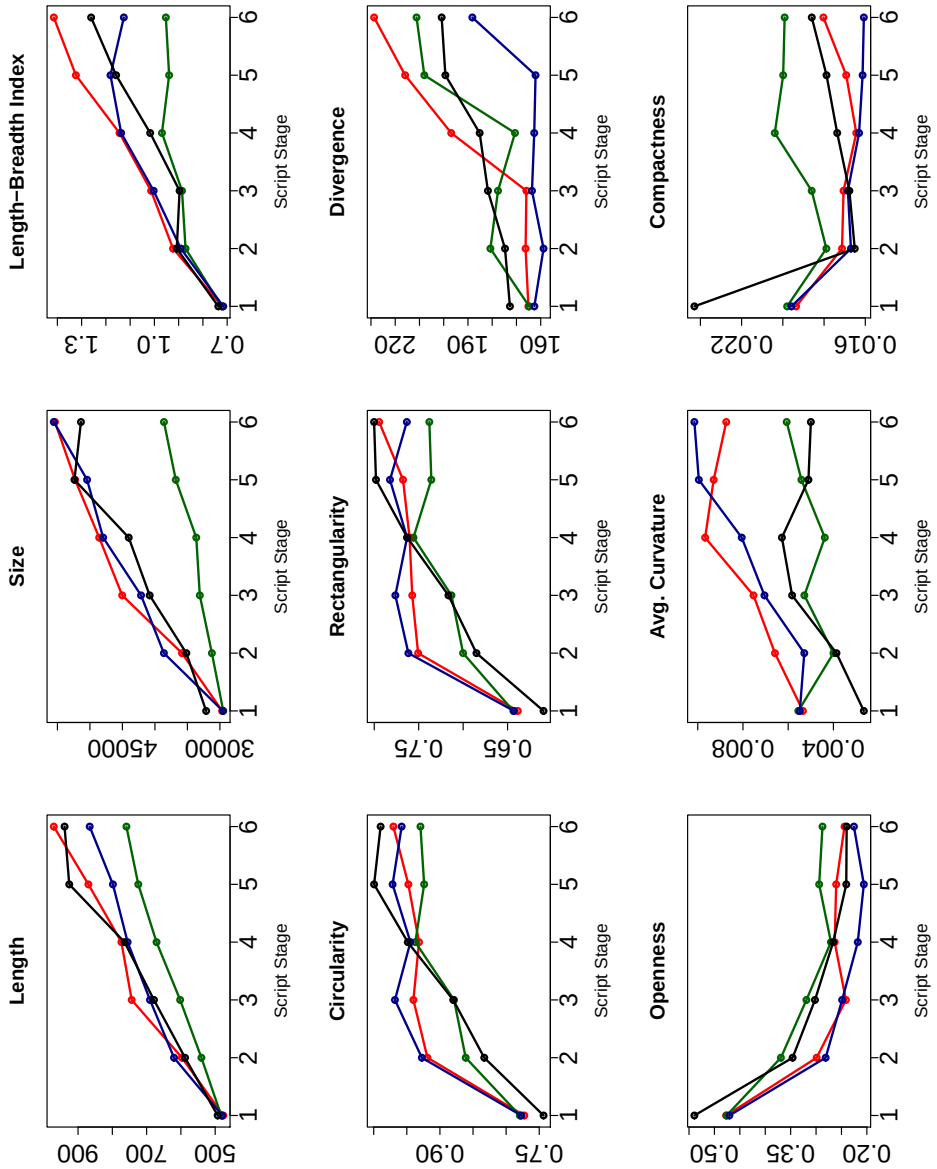


Figure 4.7: Trends in visual metrics. Red indicates *Grantha*, Green *Devanagari*, Blue *Kannada* and **Black** *Tamil*

to produce curved segments compared to straight lines (Altmann & Fengxiang, 2008), because the latter requires more effort. This may also be partially ascribed to the change in the writing surface. Brahmi was primarily an epigraphic script to be carved in stones, but as the surface changed handwriting naturalized itself, acquiring more curvature. To summarize, in terms of the visual appearance, the general trend appears to be towards *long*, *large*, *geometric*, *divergent*, *wide*, *curved*, *closed*[2] and *non-compact* characters. Even though the four scripts visually appear to be distinct, we see a common pattern of development, which is very interesting. This shows that scribal writing behaviors across India share fundamental characteristics amongst them.

Figure 4.8 shows the general trends for kinematic metrics. The split in the pen-stroke count is due to Devanagari and Kannada developing an additional pen-stroke uniformly in all characters. If this is excluded, other scripts maintain their characters as effectively requiring a single pen-stroke. The average disjoint stroke count though is seen to be increasing, fluctuating between 3.5 and 4.5. This is slightly higher than the proposed average stroke count of three by Changizi and Shimojo (2005). There also seems to be some fluctuation in retraces but at the end, it averages to one retrace per character. In terms of the length of upstrokes and downstrokes, it again shows a uniform increase as one would expect, based on the increase in the length and size of characters. Also, Brahmi starts with very low stroke changeability but as scripts develop it displays an increase. This appears to contradict the initial diversification of scripts. We can assume that such instability (due to the difference in up-strokes and down-strokes) effectively contributed the least (if at all) to diversification, with other factors probably contributing more. The entropy of writing is also seen to be increasing but tending to reach a limit ultimately. In terms of stroke metrics, the length of primitive strokes falls initially and then shows a slow growth. In terms of disjoint strokes, there is a more or less uniform increase in length. There also appears to be an increase in angles between disjoint strokes (both mean and sum) corresponding to the increase in disfluency.

[2]The term is used to indicate the lack of openness

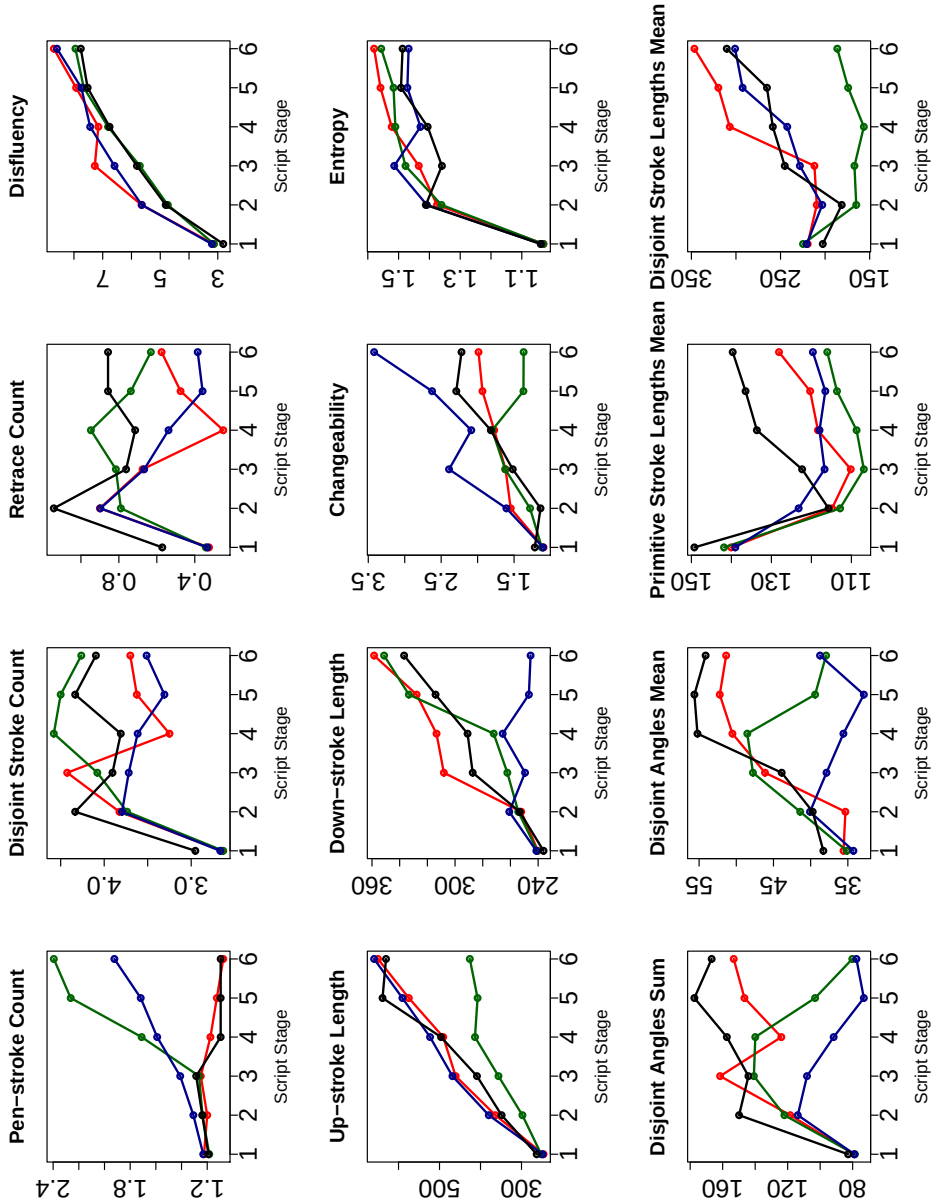


Figure 4.8: Trends in kinematic metrics. Red indicates *Grantha*, Green *Devanagari*, Blue *Kannada* and **Black** *Tamil*

In figures 4.7 and 4.8, we can see that many metrics show logarithmic or *near* logarithmic growth with compactness and openness showing a negative logarithmic growth. This shows that characters, after an initial period of diversification, begin to stabilize slowly. Explicit logarithmic growth is seen in cognitively related metrics like disfluency and entropy, which is significant. One would expect that humans tend towards reducing disfluency to increase writing speed but on a large scale it appears not to be the case. Writing appears to have gathered more disfluency, more disjoint strokes and a corresponding increase in entropy. As discussed earlier, in terms of static metrics, characters have also gained length and size as time progressed. It points to the fact that characters show a logarithmic increase in complexity in terms of production and appearance, which is counter-intuitive. Our interpretation is that this is due to *information* being continuously added, albeit in minute amounts, in terms of production and static appearance. In the end, this results in complex characters that are a product of what started out as simple geometric figures. The logarithmic profile of many metrics points to the fact that the rate of new information injected into characters slows down after some time and scripts generally tend towards stability.

4.2.2 Diversification of Scripts

Linear Discriminant Analysis (LDA) is a frequently used multivariate statistical technique to find aggregate variables that best discriminate, i.e. separate groups, in a given set of data. As a result of LDA, we obtain various discriminants that are linear combinations of variables with their coefficients corresponding to the variables' contribution to the power of that discriminant. This technique when applied to the entire script development dataset results in discriminants, comprising of major factors, that identify/label characters as belonging to a particular script. Consequently, these can be further elaborated as the factors that cause diversification and characterize scripts during the development. The datasets of all the scripts are combined into a single comprehensive set for this process, as they are more or less homogeneous. The analysis is performed separately with visual and kinematic metrics. Tables 4.1 and

4.2 show the corresponding discriminants with their coefficients.

Features	LD_V^1	LD_V^2
Length	-0.00059865341	0.007731254
Size	0.00009528834	-0.000115424
LBIndex	1.28843838989	0.186680755
Circularity	-3.48149787949	8.526610933
Rectangularity	8.27051387586	-14.373985382
Divergence	-0.00159561490	0.008506504
Openness	-0.06621315085	0.124090255
Avg. Curvature	9.73404814509	-27.513799916
Compactness	34.75293555964	-30.978259631

Table 4.1: Coefficients of visual linear discriminants

With visual metrics, we find that the first two linear discriminants - LD_V^1 and LD_V^2 - contribute around 85% of the discriminatory power. LD_V^1 discriminates scripts using mostly compactness with minor contributions from average curvature and rectangularity. LD_V^2, on the other hand, discriminates based on nearly equal contribution from average curvature and compactness, and a significant contribution from rectangularity and circularity. It follows that the scripts under discussion diversified based on the following major visual metrics - *compactness*, *average curvature*, *rectangularity* and *circularity*. We can see that characters' curvature and their shape outlines together play a major role in diversification as they particularly characterize a script. Compactness, being related to the arrangement of strokes in characters, also turns out to be one of the main factors that determine a script.

With kinematic metrics, we find that the first two linear discriminants - LD_K^1 and LD_K^2 - contribute around 72% of the discriminatory power. Though this is not very high compared to visual metrics, it is still a reasonable amount of cumulative discrimination. LD_K^1 classifies characters mainly based on *entropy*, *disjoint stroke count* and *retrace count* with minor contributions from *changeability* and *pen-stroke count*. LD_K^2 classifies mostly based on *entropy* and *pen-stroke count* with significant contributions from *disjoint stroke count* and *retrace count*. With kinematic characteristics, the investigated scripts have diversified mostly based on

Features	LD_K^1	LD_K^2
Pen-stroke Count	0.0739570676	-1.2215776695
Disjoint Stroke Count	-0.4456953605	-0.2275291566
Retrace Count	-0.2803731156	-0.1119661255
Disfluency	0.0325610130	0.0350712045
Up-strokes Length	0.0066029962	-0.0001094251
Down-strokes Length	0.0035771170	-0.0018496560
Changeability	0.0785589768	-0.1844057565
Entropy	-0.3723920326	-0.7759867001
Disjoint Angles Sum	0.0043322933	0.0084563084
Disjoint Angles Mean	0.0019477094	-0.0061158837
Primitive Stroke Lengths Mean	-0.0047329371	0.0035754377
Disjoint Strokes Lengths Mean	-0.0000481335	0.0003249647

Table 4.2: Coefficients of kinematic linear discriminants

the entropy of writing and the number of major strokes in characters of the scripts, all of which are very characteristic of writing behavior.

4.2.3 Spread of Variations

Section 4.2.1 discussed the general trends in various metrics and section 4.2.2 the overall factors driving diversification. Both of these provide a script-specific view of the script development. In this section, we analyze individual character variations that occur during each individual developmental stage. Similar to section 4.2.2, script datasets were all combined into a single set due to their homogeneity. The original feature set consisting of 9+12 metrics is too large for character-wise analysis. Hence, we proceed to perform Principal Component Analysis (PCA), which reduces our dataset and provides descriptive aggregate features. PCA is a commonly used technique in dimensionality reduction to reduce

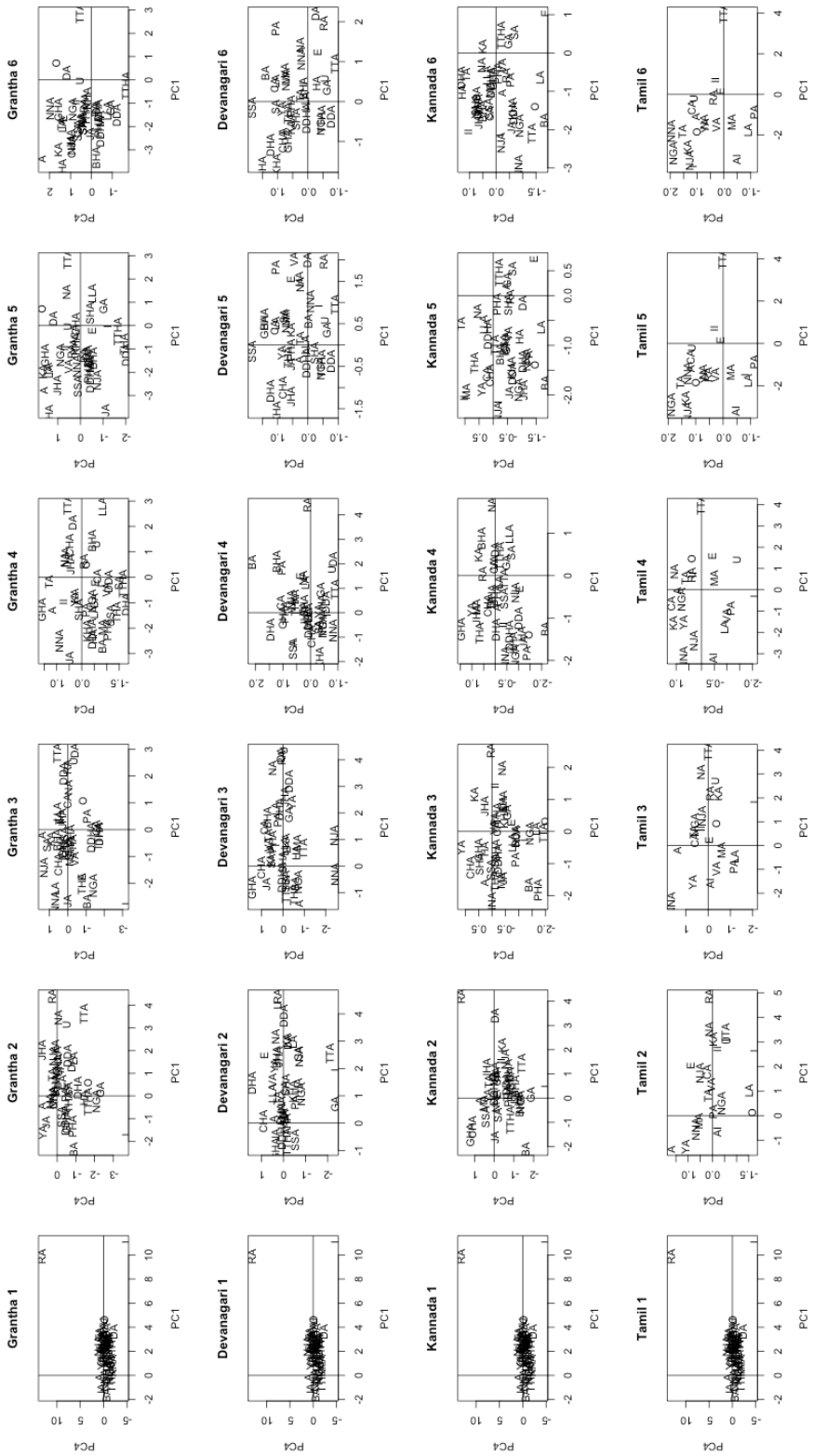


Figure 4.9: Plot of PC1 vs PC4 for scripts in all 6 stages of development. The labels for the data points refer to the Unicode names of characters

a multivariate set to a small set of aggregate variables called *Principal Components* that describe the same set effectively by accounting for most of the variance. These principle components uncover the salient features of a dataset by revealing relationships among variables.

Feature	PC1	PC2	PC3	PC4
Size	-0.324	0.493	-0.300	0.171
LBIndex	-0.244	0.253	0.299	
Circularity	-0.478	-0.245	0.274	-0.246
Rectangularity	-0.463	-0.269	0.348	-0.237
Divergence		0.485	0.512	0.230
Openness	0.350	0.256	0.526	-0.139
Avg. Curvature	-0.204	0.261	-0.131	-0.473
Compactness		-0.399	0.244	0.597

Table 4.3: Loadings of principal components

Table 4.3 shows the first four principal components derived by applying PCA to the visual metrics in our dataset. The principal components shown account for nearly 78% of all variance and hence can be considered satisfactory to abstract the dataset. Principal components are usually interpreted as a comparison between the negative and positive loadings with the magnitude and sign of a principal component being indicators for the influencing factors.

PC1 is a comparison between openness, and mainly circularity & rectangularity (which can together be taken to indicate geometricity) and length. Characters that are *open*, *short* and *non-geometric* have positive scores, while *closed*, *long* and *geometric* characters will have large negative scores. PC2 compares compactness, circularity and rectangularity with mostly size and divergence. Characters with negative PC2 scores are typically *compact* and very *geometric*. Positive scores indicate characters that are *large*, *non-compact* and *divergent*. For PC3, high negative scores indicate *large* and highly *curved* characters. For PC4, large negative scores point to highly *curved* and *geometric* characters with positive scores pointing to characters that lack those characteristics.

We specifically discuss the plots of PC1 vs PC4 for illustration. It can be clearly seen from figure 4.9 that Brahmi characters initially have very

similar visual profiles (evident by the crowded overlap of characters). But as time passes, the characters diverge significantly as discussed earlier. Interestingly, we can see a particular pattern in the diversification process. In Brahmi, the characters are primarily around the first and fourth quadrant boundary. Here, the characters are just *open*, *short*, and *non-geometric*. However, during the second stage of diversification, characters gain more *geometricity*, *closure*, and *length* moving towards other quadrants but mostly dispersing towards the first and third quadrants with ultimately many of the characters moving into the second and third quadrants gaining *curved geometricity* along with *lengthy closure*. We can clearly see the interplay of features that cause variations in handwriting and a specific pattern in which characters have developed.

Similarly, we could perform analysis on the kinematic metrics as well.

4.2.4 Interaction Between Metrics

To explore the interaction between various metrics, a regression analysis is performed. Similar to sections 4.2.2 and 4.2.3, individual scripts datasets were combined for this process. However, we ignore the initial stages of the scripts as we think it is unlikely that metrics would have interacted then. We discuss below the relevant correlations that were noticed in this analysis. See table 4.4 for the scale of strengths of correlations.

Absolute value of r	Strength
0.00 - 0.19	Very weak
0.2 - 0.39	Weak
0.4 - 0.59	Moderate
0.6 - 0.79	Strong
0.8 - 1.0	Very strong

Table 4.4: Scale for the strengths of correlations

Length and size have a positive correlation with disfluency (r=0.74) and entropy (r=0.47). This can be attributed to the fact that characters become more complex as they gain length and size, as seen earlier in section 4.2.1. The gain in length is mostly in terms of up-strokes (as

demonstrated by the very strong correlation coefficient of 0.87 between the two metrics). Retraces appear less frequently with the increase in curvature (r=-0.49). This is reasonable, as retraces are generally difficult with cursive strokes. Average curvature also shows a moderately negative correlation (r=-0.54) with disjoint stroke count as characters with higher curvature have fewer disjoint strokes and vice versa. Also, a break in pen movement likely contributes to a reduction of curvature as shorter strokes tend to be less curved. This can be clearly seen by the moderately positive correlation (r=0.66) between the average length of disjoint strokes and curvature. Retraces also appear to occur frequently with characters that have more disjoint strokes (r=0.8). The more disjoint a character is, the greater the chance that it requires retraces to complete the trajectory. Stroke lengths appear to have a moderately negative correlation with the stroke counts as breaks in trajectory most likely reduce the length. It is evident that the breaks (i.e. disfluency) visually affect characters in different ways as seen by the above results.

4.2.5 Impact of Frequency on Handwriting

The usage of characters is invariably tied to the language they represent. Given this, it would be reasonable to assume that frequency of characters might have had some influence on the development of scripts. In our case study, we only take monolingual scripts, namely Tamil and Grantha, into consideration. Kannada and Devanagari were bilingual during their development, representing both Sanskrit and the local language. It would be overly complex to predict the influence of two different languages on the scripts and hence the restriction to discussing monolingual scripts. For the Grantha script, the frequency of characters in Classical Sanskrit was used and modern Tamil data was used for the Tamil script. Given that Tamil and Sanskrit form a linguistic continuum over time, we assume that the frequency data for these two languages could have remained more or less stable, at least with regard to their core phonemes. Similar to the previous section, we also ignore the initial prototype stage of the scripts from the analysis and consider only the developmental stages, where frequency would have had some interaction. For the list of correlation

values see tables 4.5 and 4.6. Also refer table 4.4 for the scale of correlation strengths.

Our regression analyses of various metrics with character frequencies show no strong correlation between frequency and character metrics. However, we do see some low to moderate correlations. For instance, disfluency and entropy appear to have a weak negative correlation with frequency in most cases. Even with visual metrics, size and length show a moderate to low negative correlation with frequency. This can be explained by the fact that frequently used characters tend more towards minimizing the kinematic effort. Similar results occur in Chinese writing, where character frequency was found to be inversely proportional to the number of strokes (Gao & Kao, 2002). The effect of frequency appears to be more pronounced in Tamil than in Grantha. Also, the scripts in their intermediate stages show weaker correlations but in their last stage, their correlations are generally stronger. Particularly, intermediate Grantha developments show very low negative correlation to no correlation with respect to size and length. This is perhaps due to the effects of frequency being accumulated towards the final stage of development.

	Disfluency	Entropy	Size	Length
Tamil 2	0.09	0.25	-0.04	-0.25
Tamil 3	-0.18	-0.36	-0.12	-0.20
Tamil 4	0.09	-0.27	-0.18	-0.41
Tamil 5	-0.212	-0.03	-0.15	-0.29
Tamil 6	-0.57	-0.42	-0.40	-0.36

Table 4.5: Regression coefficients for Tamil

	Disfluency	Entropy	Size	Length
Grantha 2	-0.16	-0.13	-0.05	0.01
Grantha 3	-0.28	-0.21	-0.03	-0.14
Grantha 4	-0.27	-0.39	0	0
Grantha 5	-0.14	-0.05	0.05	0
Grantha 6	-0.32	-0.06	-0.31	-0.32

Table 4.6: Regression coefficients for Grantha

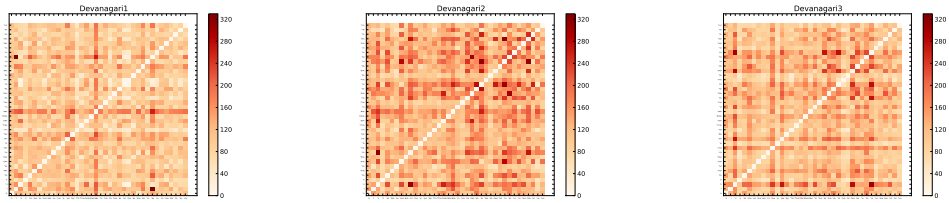
The absence of any strong correlations and the presence of weak or moderate correlations indicates that instead of a having direct impact on characters, the frequency of usage has a more formative background influence in the development. This is an indication that character frequency could have affected scribal behavior.

4.2.6 Internal Distinctiveness

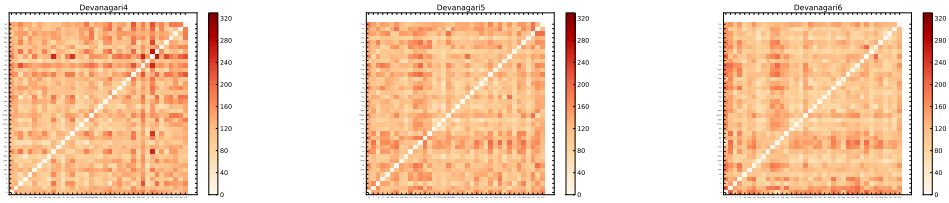
Our case study now addresses the change in the internal distinctiveness of scripts to evaluate the similarity of characters within a set over time. Accordingly, we construct a self-similarity matrix of characters in the scripts using the distinctiveness values that are computed by applying DTW to their static shapes. The distinctiveness values used to construct the matrix are then normalized based on the length of characters to avoid penalizing characters based on their size. The normalization adjusts the distinctiveness of longer characters that will otherwise have disproportionately high distinctiveness values (Serva & Petroni, 2008). Using this self-similarity matrix, we then proceed to generate heat maps for scripts as shown in figures 4.10, 4.11, 4.12 and 4.13. The heat maps are all calibrated to the same scale to facilitate straightforward visual comparison. Lighter shades indicate low distinctiveness, while the darker shades indicate a higher value.

We observe a common tendency in the evolution of distinctiveness of the scripts. We can see that in the first stage, the characters are fairly similar. During the next stage, they appear to gain some distinctiveness but slowly move towards normalizing the shapes as seen with the progressively lighter shades of the heat maps. The initial change is what caused the initial impetus for diversification of the source script into various other scripts as noted in the previous sections. This was probably caused by the adaptation of Brahmi (which was designed to be carved in stones) into various materials for day-to-day use by different people. But as self-contained sets, scripts appear to normalize their shapes by becoming more and more self-similar over time. We saw a similar phenomenon of stabilization in section 4.2.1 as well.

The previous sections analyzed overall features regarding the changes

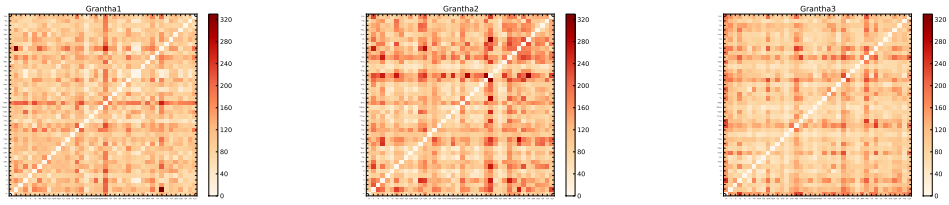


(a) Devanagari 1 (b) Devanagari 2 (c) Devanagari 3

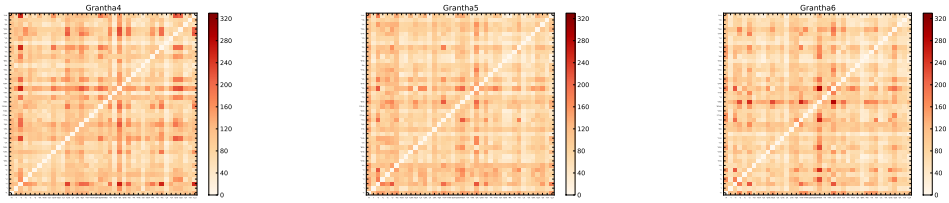


(d) Devanagari 4 (e) Devanagari 5 (f) Devanagari 6

Figure 4.10: Devanagari distinctiveness

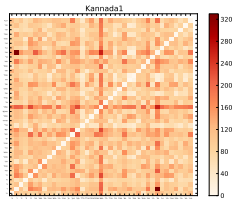


(a) Grantha 1 (b) Grantha 2 (c) Grantha 3

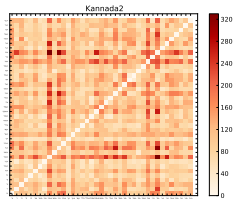


(d) Grantha 4 (e) Grantha 5 (f) Grantha 6

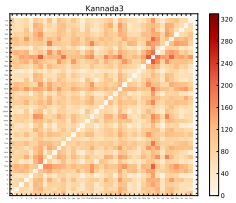
Figure 4.11: Grantha distinctiveness



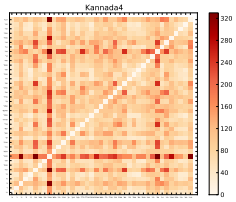
(a) Grantha 1



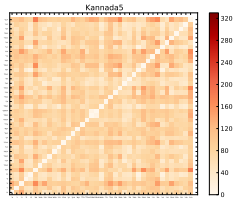
(b) Grantha 2



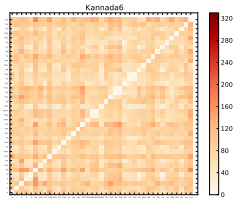
(c) Grantha 3



(d) Grantha 4

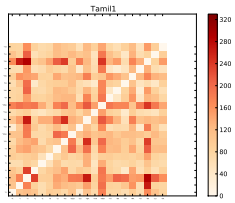


(e) Grantha 5

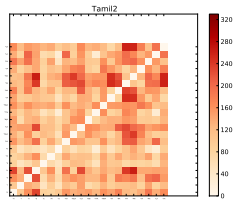


(f) Grantha 6

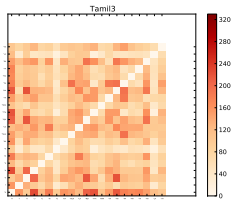
Figure 4.12: Kannada distinctiveness



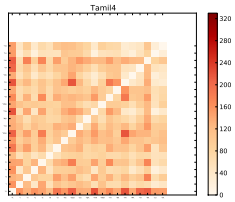
(a) Tamil 1



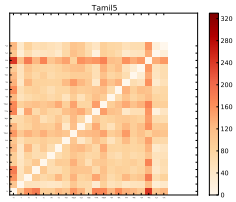
(b) Tamil 2



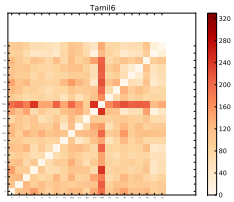
(c) Tamil 3



(d) Tamil 4



(e) Tamil 5



(f) Tamil 6

Figure 4.13: Tamil distinctiveness

in handwriting behavior. In the following subsections, we attempt to perform a stroke-level analysis of our dataset.

4.2.7 Writing Perplexity

As discussed in section 3.2.5, we construct a bigram model of scripts by translating the trajectory of characters into direction codes. To calculate the perplexity of a script, we calculate the perplexity of each character in the script by using the rest of the characters as a training set for the bigram model. This results in a list of perplexities, one for each character. The geometric mean of those perplexities is taken to be the perplexity of the script. We did consider selecting a random subset of characters as testing data, but realized that given the small set we were dealing with, it would inadvertently result in a sample bias. Therefore, an average value appeared to be a better indicator for an entire script.

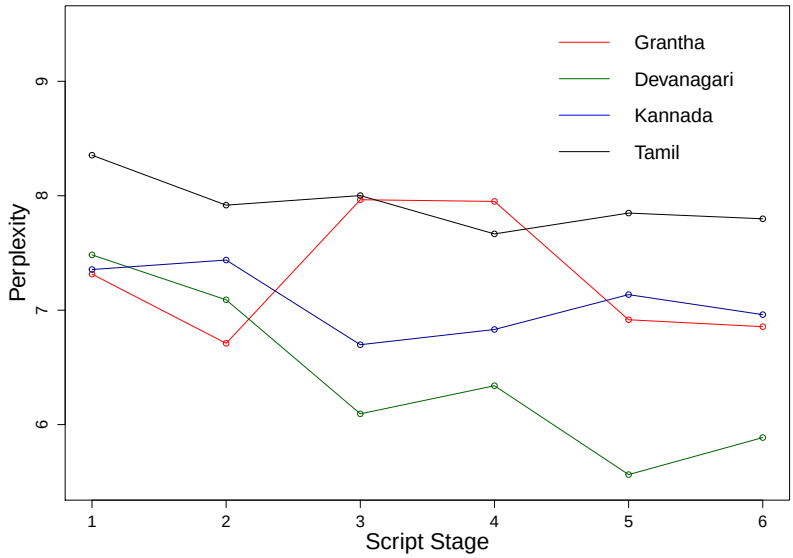


Figure 4.14: Changes in script perplexity

From figure 4.14, we can observe that the perplexities of the scripts under consideration are generally showing a decreasing trend. This indicates that most scripts have progressively become homogenized in terms

of trajectory. This means that certain trajectory patterns were normalized and repeated in other characters within a script. As we just saw in section 4.2.6, the same behavior is seen for visual appearance as well (see section 4.2.8 as well). Although the four scripts share the same trend, there is a small anomaly with the increase in perplexity from the second stage of the Grantha script to the third stage. This can be attributed to the fact that Grantha at that stage faced a comparatively large number of stroke augmentations. This may have resulted in abrupt changes in the trajectories, which is represented as high perplexity. However, it is to be noticed that after the spike, perplexity eventually settles down to a lower value.

4.2.8 Stroke Repertoire

To properly analyze the development of Indic scripts, it is necessary to study the changes in their stroke repertoire. The script repository that we created already had characters segmented into disjoint strokes and primitive strokes. We extract the corresponding stroke repertoire for the scripts using the prototype of our framework. Figures 4.15 and 4.16 show the repertoire as extracted by the prototype for disjoint and primitive strokes respectively. We impose an empirical DTW cost threshold to decide if two strokes are to be considered the same and calculate the minimal set. This may not have resulted in an ideal minimal set, but it can be considered to be a close approximation subject to some error. We present below the trends in the change of script repertoire count during the script development.

We can see in figure 4.17 that there is a steady increase in the count of disjoint strokes followed by a steady decrease. This trend is not very pronounced in Tamil, probably because of its smaller character set (20 compared to 40 in other scripts). This trend can be interpreted as another stabilizing mechanism. Beyond a certain threshold, the scripts are likely to normalize themselves visually as noted in the previous sections, ultimately resulting in fewer patterns. The stabilizing behavior appears to be quite prominent during script development. We demonstrated such behavior in sections 4.2.1, 4.2.6 and 4.2.7. In terms of primitive

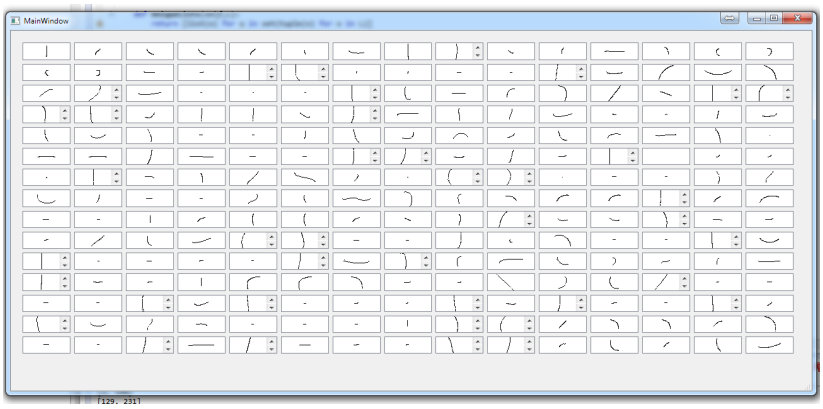


Figure 4.15: Disjoint stroke repertoire for Grantha 3

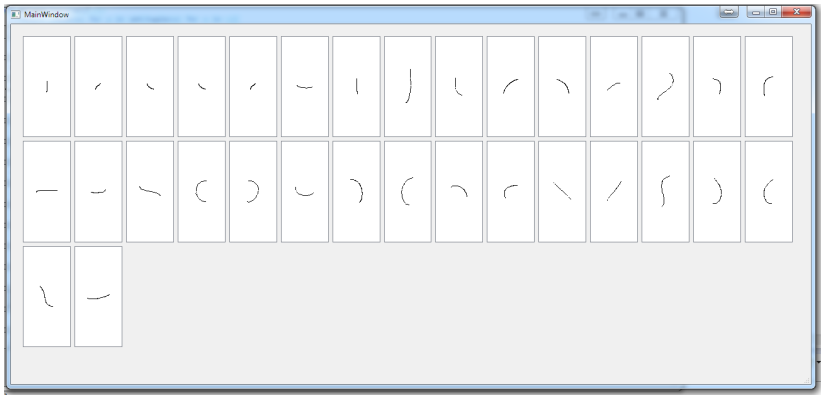


Figure 4.16: Primitive stroke repertoire for Grantha 3

strokes, as displayed in figure 4.18, we see a fairly stable behavior with slight fluctuation. This points to the fact that primitive strokes have remained fairly constant with only their combinations giving rise to different composite strokes.

We also perform a regression analysis of the disjoint stroke counts and primitive stroke counts with that of perplexity. With the primitive strokes, we found a weak positive correlation (r=0.36) with perplexity. This is understandable as more strokes add to the *confusion* of selecting pen movements when writing a character. However, with the disjoint strokes, we found that there was a weak negative correlation (r=-0.32). We assume this to be the case because a higher number of composite strokes is probably also accompanied by an increase in the frequency of

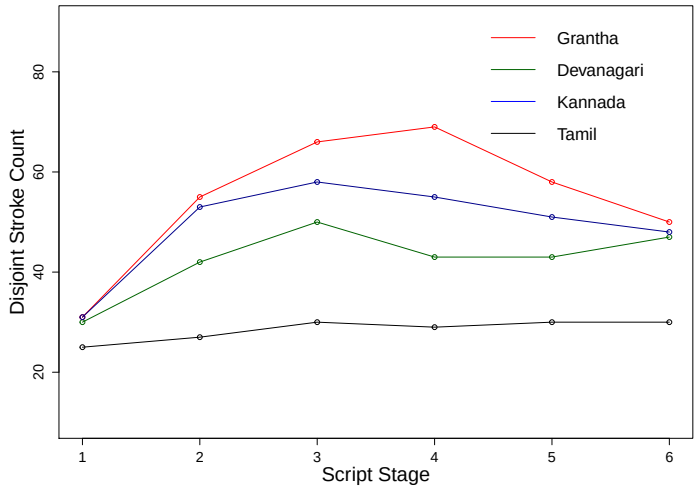


Figure 4.17: Trends in disjoint stroke repertoire count

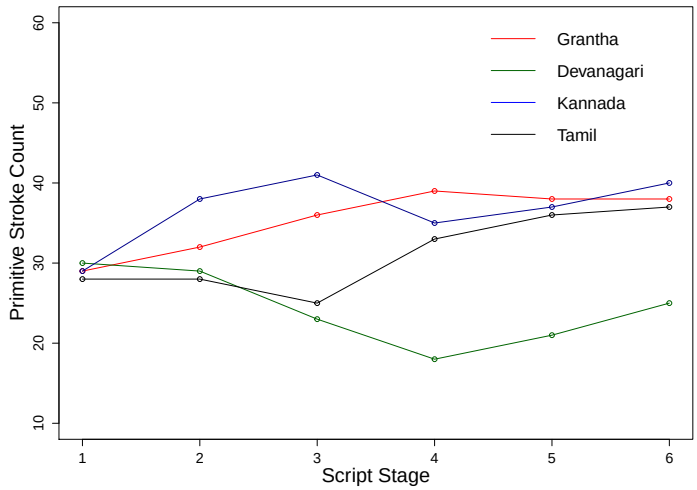


Figure 4.18: Trends in primitive stroke repertoire count

sub-trajectory patterns.

4.2.9 Major pen directions

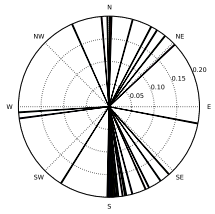
We have visualized the major pen directions of the four different scripts in figures 4.19, 4.20, 4.21 and 4.22. As seen in these figures, strokes in

the downward direction (not necessarily down-strokes though) appear to be the major part of most scripts' characters. In some cases like Grantha and Kannada, as the scripts evolve rightward strokes appear to dominate more (which is expected for a left-to-right script). The paucity of mostly upward strokes in the scripts can be attributed to the writing material. Palm leaf manuscripts were more likely to be damaged by carving upward strokes and hence the scripts possibly developed predominantly with downward strokes in their stroke repertoire.

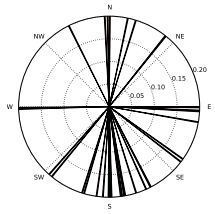
4.3 Handwriting Modeling of Indic Script Development

In the previous sections, we focused mostly on various metrics and stroke counts, and indirectly linked their association with changes in scribal behavior. However, we did not directly analyze the trajectory changes themselves. This leads us to handwriting modeling of the script development. As mentioned in section 4.1, the scripts under consideration were mostly written on palm leaves (with an iron stylus) and on dried birch-bark (with reed-pens using ink). The writing process on those materials is not fundamentally different from that modeled by the Sigma-Lognormal model. Hence, we can consider the model to be appropriate for analyzing changes in our case study. We show that through handwriting modeling, scribal behavior can be studied in finer detail and can quantify various stroke phenomena that cause shape changes.

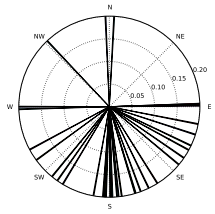
We experiment with handwriting modeling for two different datasets - Brahmi (*Tamil 1*) to very early Tamil (*Tamil 2*) and Brahmi (*Kannada 1*) to very early Kannada (*Kannada 2*), which covers the divergence of both very early Tamil and Kannada from Brahmi (see figure 4.23). We use handwriting modeling techniques to investigate the changes to scribal behavior that lead to this split. We begin by constructing Sigma-Lognormal models of characters in our data as shown in section 3.3.3. We then model the shape changes similarly to section 3.3.4. We initially attempted to perform automated morphing of the character shapes using genetic



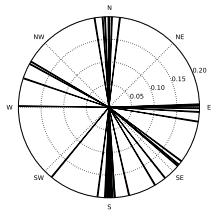
(a) Devanagari 1



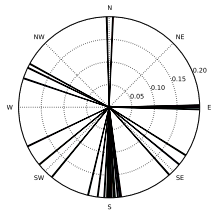
(b) Devanagari 2



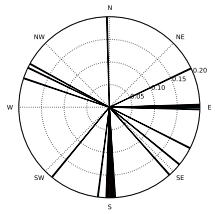
(c) Devanagari 3



(d) Devanagari 4

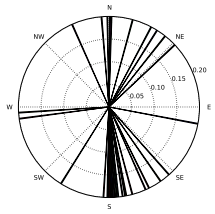


(e) Devanagari 5

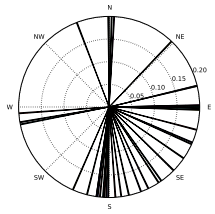


(f) Devanagari 6

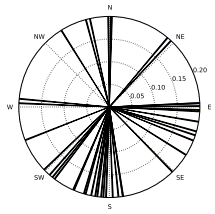
Figure 4.19: Devanagari stroke directions



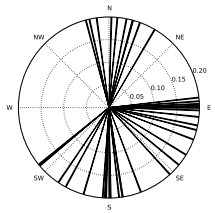
(a) Grantha 1



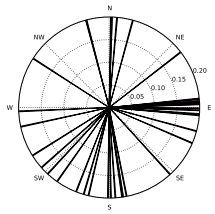
(b) Grantha 2



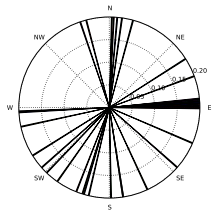
(c) Grantha 3



(d) Grantha 4

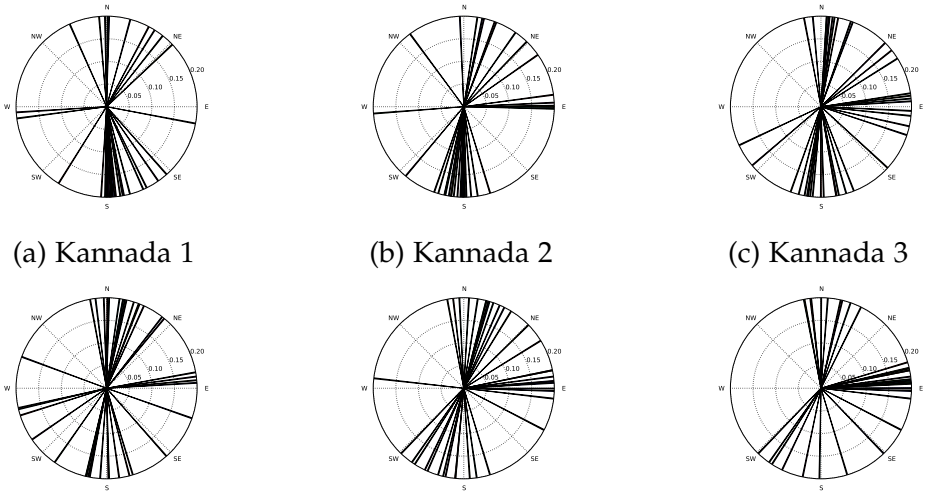


(e) Grantha 5



(f) Grantha 6

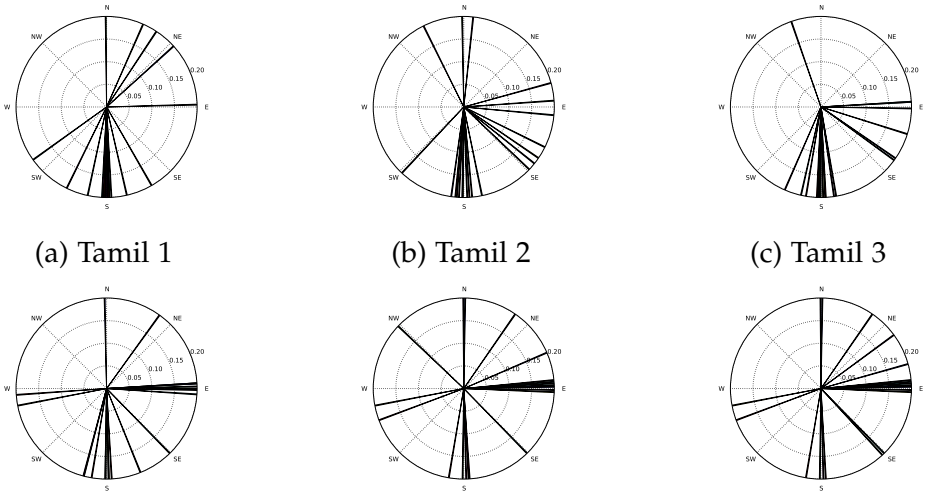
Figure 4.20: Grantha stroke directions



(a) Kannada 1 (b) Kannada 2 (c) Kannada 3

(d) Kannada 4 (e) Kannada 5 (f) Kannada 6

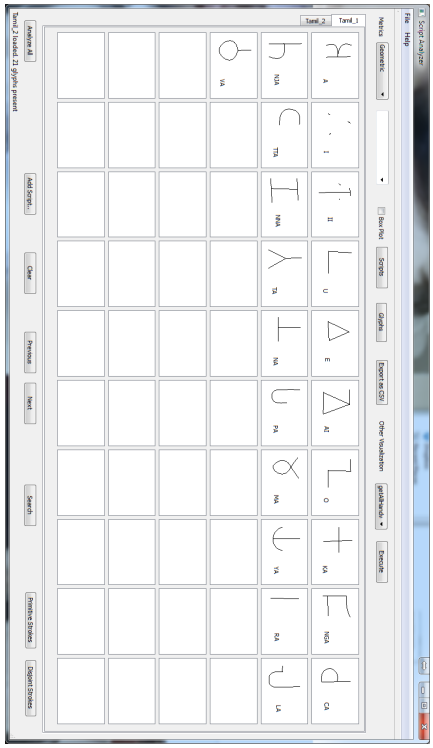
Figure 4.21: Kannada stroke directions



(a) Tamil 1 (b) Tamil 2 (c) Tamil 3

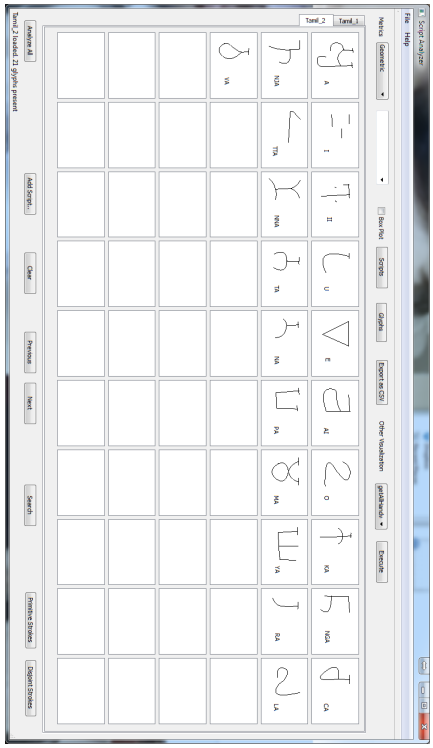
(d) Tamil 4 (e) Tamil 5 (f) Tamil 6

Figure 4.22: Tamil stroke directions

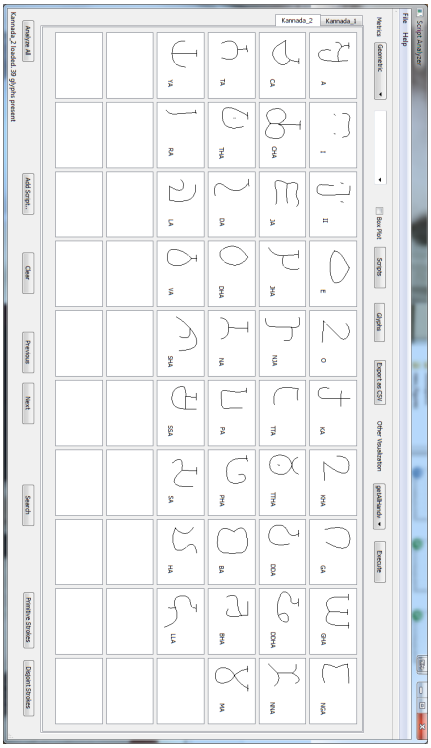


(a) Repository for Tamil 1

(b) Repository for Tamil 2



(c) Repository for Kannada 1



(d) Repository for Kannada 2

Figure 4.23: Script repositories of Tamil & Kannada

algorithms, but this did not yield usable results. We concluded that this was due to the chosen cost function (DTW) and the solution space of the problem. Even though DTW is intuitive for humans, it appears to be inappropriate for genetic algorithms. Also, assuming a character has an average of four strokes and each stroke is represented by four parameters, morphing a character has approximately 16 degrees of freedom (4×4), which results in a huge solution space. Finally, we resorted to manual tuning to make the process easier and quicker. The Sigma-Lognormal parameters were changed manually to administer the shape change of characters using the following order of precedence:

1. The time parameter (t_0)
2. The angle parameters (θ_s & θ_e)
3. The length parameter D

Then by studying the difference in the parameters of characters through their transformation matrices, we try to interpret the differences in terms of scribal behavior. In most cases, we manipulate the parameters to effect the shape change of Brahmi characters into Tamil and Kannada characters. In some cases, it was easier to morph Tamil and Kannada characters back to Brahmi. As explained in section 3.3.2, we just negate the resultant transformation matrix of this reverse development to explain the shape change in the forward direction. Some characters have additional stroke augmentations during the development. Even though these could have been explained via reverse development as discussed earlier, we choose not to include them to simplify our analyses. As such, the below sections are limited to discussing stroke phenomena of the (pre-existing) constituent strokes (and not newly augmented strokes).

4.3.1 Individual Stroke Phenomena

The individual effect of the three basic phenomena on characters' strokes, namely the change in length (through D), the change in stroke timings (through t_0) and the change in curvature (through $\Delta\theta$) is studied here. The following percentages of change in strokes were calculated by imposing

a threshold on the transformation matrices to ignore very minor changes in the Sigma-Lognormal parameters. If a stroke's parameter in the transformation matrix was found to be beyond this threshold, it was considered as to have been changed with respect to that parameter.

Figures 4.24 and 4.25 show histograms of the magnitude of change in stroke lengths in the development of Tamil and Kannada respectively. For the sake of readability, we ignore stable strokes in the plots. Any difference greater than 10% of the original stroke length is considered to be significant by us. In the case of Kannada, only 25% of the overall strokes have some change with respect to the length, with 20% of the overall strokes showing significant change. Amongst them around 55% show increase in length while the other 45% have decreased. Comparatively, Tamil shows a higher rate of change in length with around 35% of overall strokes showing change. But only 17% of these strokes show significant change, out of which 55% show increase and 45% show decrease. It is seen that Kannada overall has gained length compared to Tamil. From the figures, it can also be seen that the strokes within characters have been consistent with the type of change.

Figures 4.26 and 4.27 show changes in curvature of strokes. As in the case of stroke lengths, we do not show stable strokes. 55% of the strokes in Tamil show curvature change as opposed to 30% of strokes in Kannada. As seen earlier, strokes within the same character exhibit similar patterns of change. With respect to stroke timings (see figures 4.28 and 4.29), nearly a quarter of strokes in Tamil and Kannada show change, with strokes showing a partial preference to decrease in stroke timings (leading to smoother strokes) in both the cases.

In total around 70% of strokes in Kannada show change (in one or more aspects), but in Tamil, this increases to 90%. Apparently, Tamil has undergone a drastic change in writing behavior compared to Kannada. This is reasonable considering the fact that Tamil had a thriving regional scribal tradition (as supported by the numerous epigraphs) and thus would have been more inclined to adapt (and hence inadvertently) change the Brahmi script.

With respect to factors influencing the shape change, it is quite evident (see figure 4.30) that Tamil has a higher proportion of change through

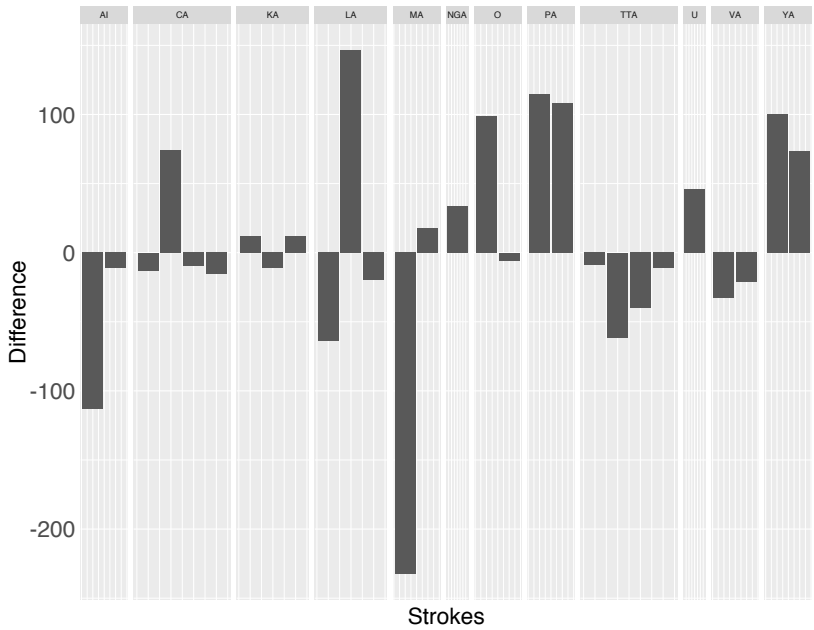


Figure 4.24: Change in stroke length during the development of Tamil

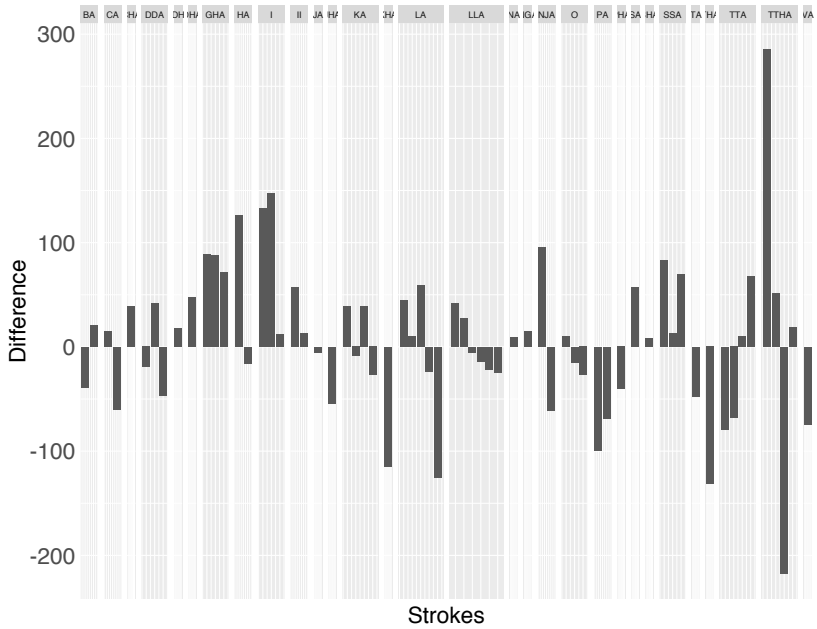


Figure 4.25: Change in stroke length during the development of Kannada

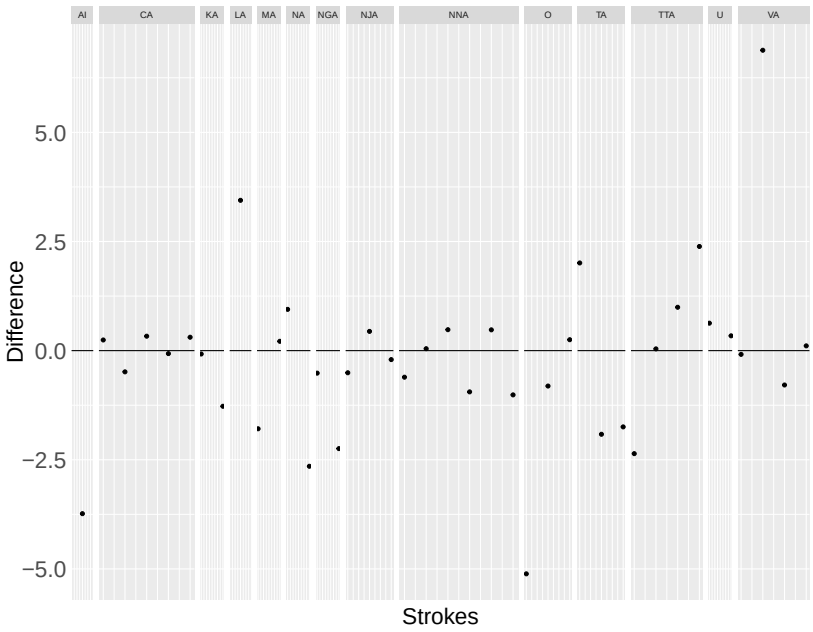


Figure 4.26: Change in curvature angle for Tamil

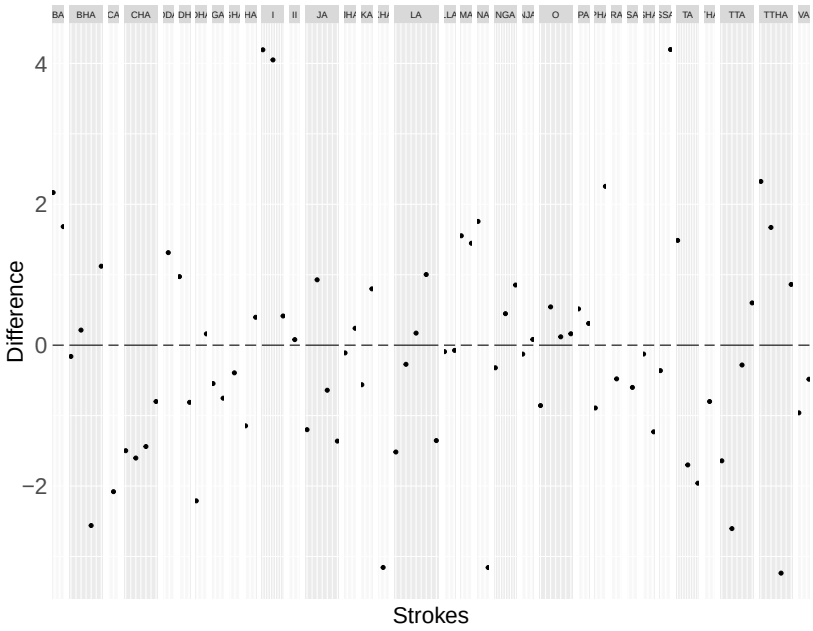


Figure 4.27: Change in curvature angle for Kannada

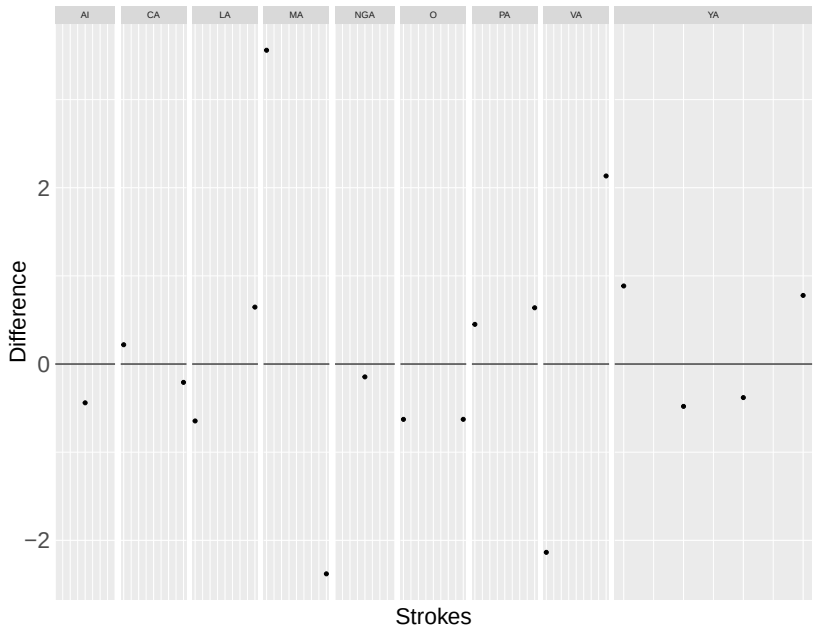


Figure 4.28: Change in stroke time for Tamil

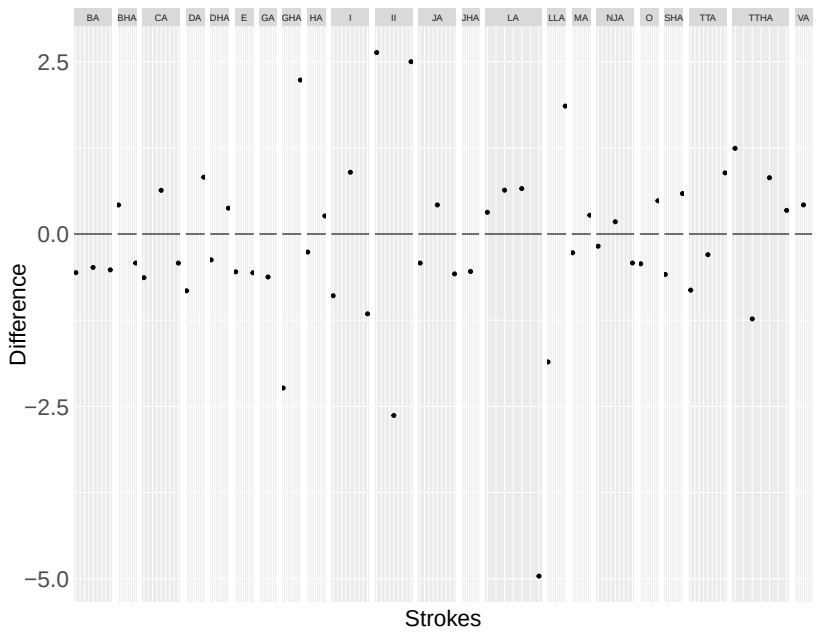


Figure 4.29: Change in stroke time for Kannada

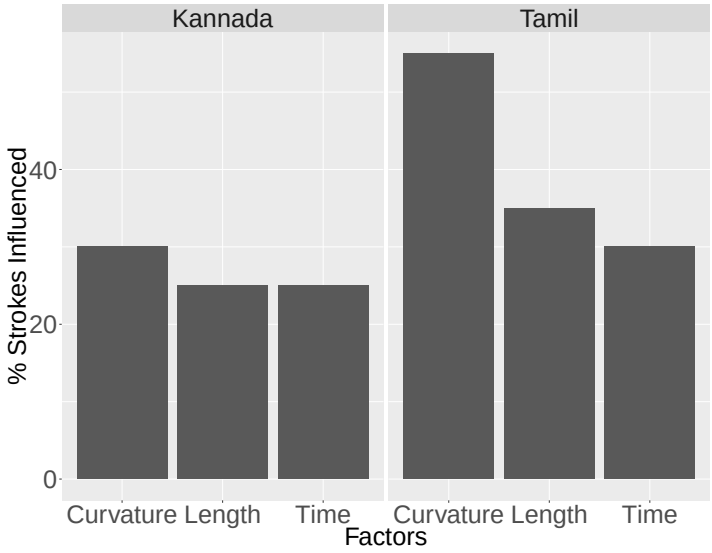


Figure 4.30: Factors influencing stroke change

curvature. This is not surprising as this early form of Tamil script is actually named *Vaṭṭeḻuttu* (Round Script). In Kannada, all the three factors appear to have more or less the same level of influence.

4.3.2 Interplay Between Stroke Phenomena

While we present the influence of individual stroke phenomena in the previous section, what interests us more is the interplay between them that gives rise to the split between Tamil and Kannada from the original Brahmi script.

Figure 4.31 shows the parallel plots of the transformation matrices obtained from transforming Brahmi characters to Tamil and Kannada. It can be seen that even though Tamil and Kannada appear to share some traits in changes, there are also several noticeable differences. These phenomena are responsible for the similarity and at the same time the contrast between the early forms of the two scripts. In fact, Tamil appears to have two different types of distinct behaviors and Kannada three, and all these variants appear to be related to the starting angle of strokes. The simultaneous interaction between the three phenomena can also be visualized as a 3D plot as seen in figure 4.32. But three-dimensional

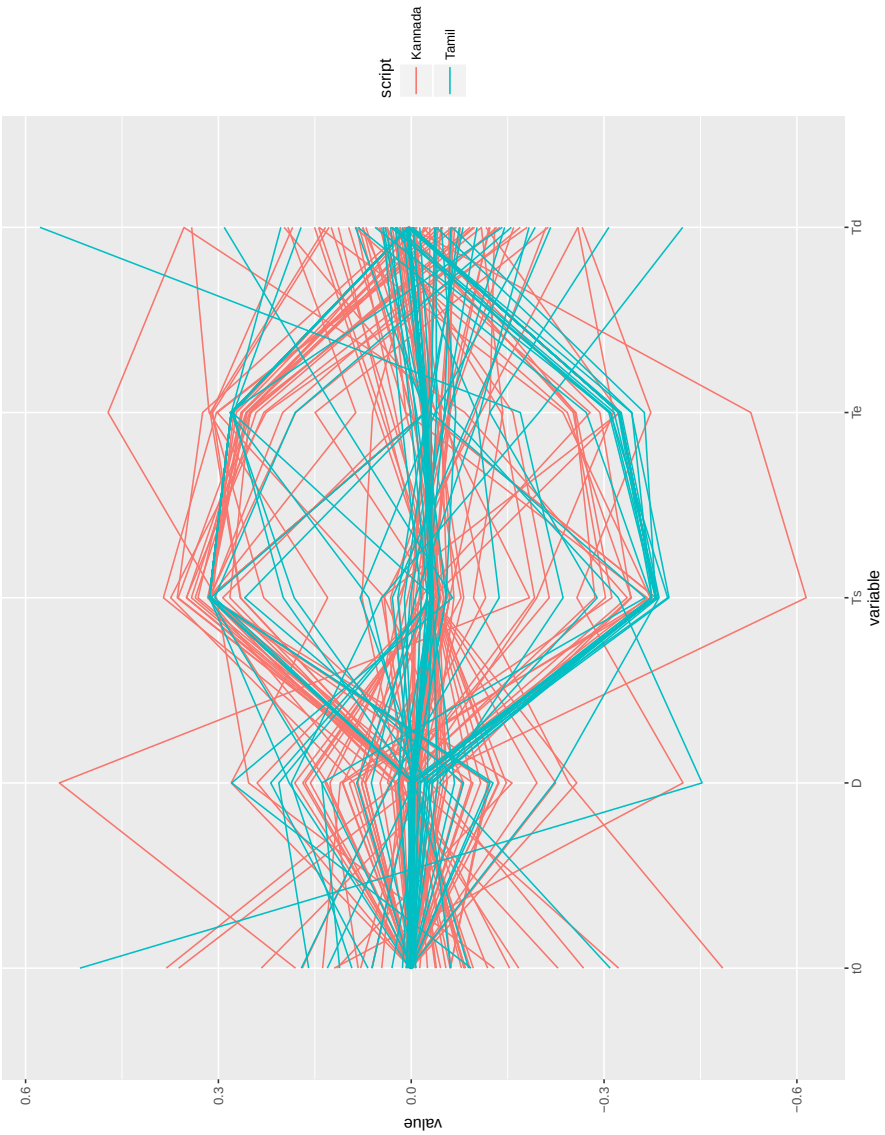


Figure 4.31: Parallel plot of various stroke parameters for Tamil and Kannada. Ts refers to θ_s, Te to θ_e and Td refers to $\Delta\theta$.

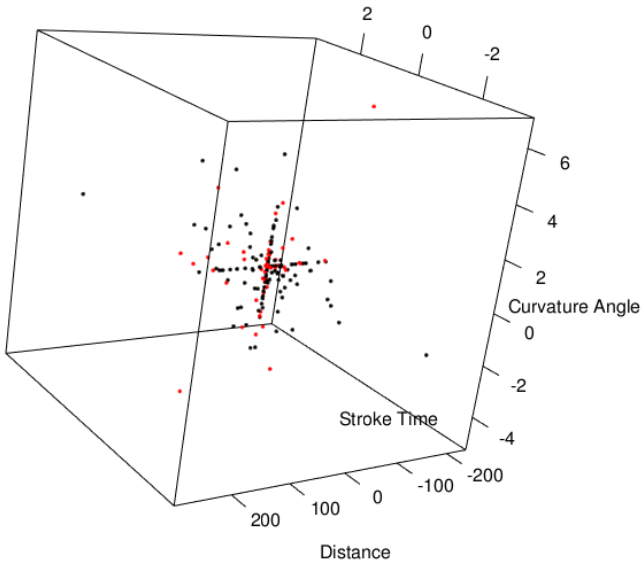


Figure 4.32: 3D plot of the parameters. Red color denotes Tamil and Black denotes Kannada

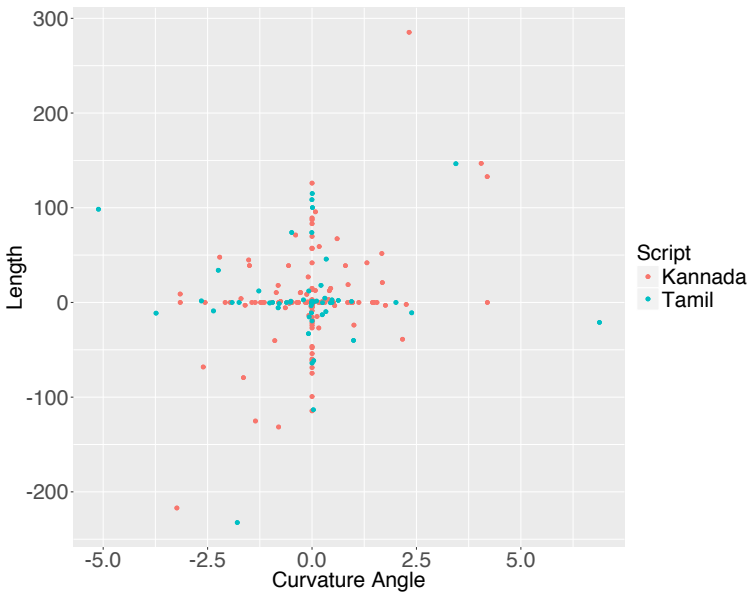


Figure 4.33: Stroke Curvature vs Stroke Length

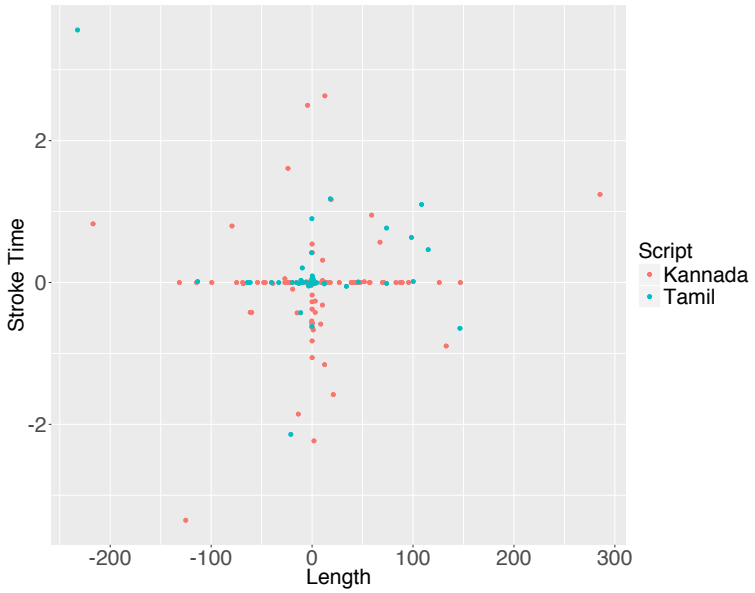


Figure 4.34: Stroke Length vs Stroke Time

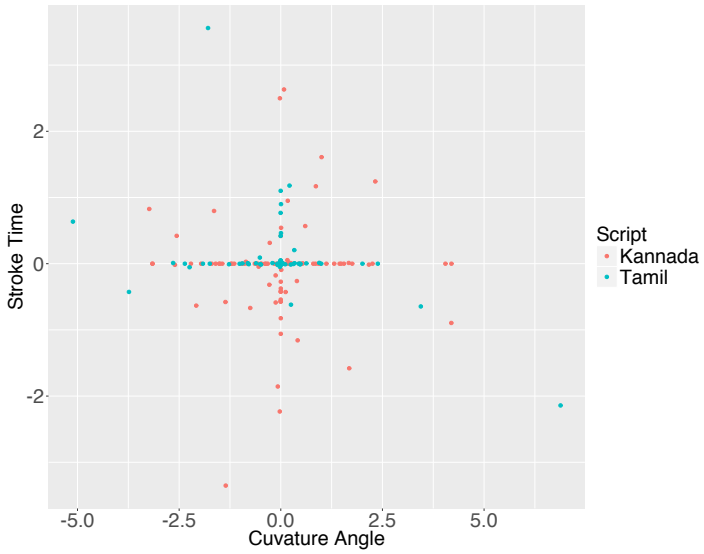


Figure 4.35: Stroke Curvature vs Stroke Time

graphs are non-intuitive and hard to interpret(Wright & Williams, 2005). Therefore, we visualize the interplay of stroke phenomena in pair-wise combinations as shown in figures4.33, 4.34 and 4.35, which helps us to interpret the interactions better.

In the development of early Kannada, strokes appear to exhibit a simultaneous change in curvature and distance as opposed to Tamil, where strokes mostly exhibit only either one of those behaviors. Figure 4.33 shows most Tamil characters lying completely on the x and y axes compared to Kannada whose characters are seen spread across the quadrants. This is also seen through correlation between the absolute magnitude of change in distance and curvature. For Kannada this is 0.4, whereas for Tamil, it is 0.2. Specifically, the change towards positive curvature angles in Kannada seems to have some influence on the length of strokes. Comparatively, stroke timing and length appear to have a moderate concurrent effect. Ligation in strokes is usually followed by the change in length in Kannada as seen in figure 4.34 but seldom in Tamil. The smoothening of strokes evidently cause the decrease in stroke length (due to strokes being merged). Perhaps, the scribes were trying to compensate the decrease by further increasing the length of strokes. Similarly, Kannada appears to have more strokes where ligation also affects the curvature compared to Tamil (see figure 4.35). It is probably for aesthetic reasons that scribes consciously try to curve the strokes as they smoothen them. Overall, early Kannada appears to be the result of several concurrent stroke phenomena in the handwriting process compared to Tamil, where individual stroke phenomena appear to have had more influence. In fact, as seen earlier, curvature appears to be the main factor behind the development of early Tamil. It is interesting to note how the interaction between stroke phenomena (or the lack thereof) appears to have resulted in the divergence of Brahmi into Kannada and Tamil.

4.4 Open Data

Following our decision in section 3.2.6 to release the source code of our framework prototype under an open source license, we also release

the data used in the various analyses presented in this chapter under the same license. This release includes all the script repositories (to be used with our framework) and the quantified metrics (as CSV files). The R scripts that were used to perform quantitative methods on the dataset are also available in the repository. In terms of handwriting modeling, we release all the raw handwriting signal files, the extracted Sigma-Lognormal models and the final manipulated models to explain the shape changes. They are all available in the following URL:

https://github.com/virtualvinodh/indicdevanalysis

We hope this will enable researchers to explore the dataset with further detailed analyses.

4.5 Summary

We presented the development of Indic scripts as a suitable case study for illustrating the effectiveness of our framework. The framework was used to analyze characters by performing a range of analyses and the corresponding results with interpretations were presented. The results showed that the framework can effectively describe both overall and individual stroke behavior through quantitative methods. Furthermore, we demonstrated the use of handwriting modeling to study the change in handwriting behavior by applying it to the divergence of Brahmi into Tamil and Kannada. We provided access to our data as a part of making our framework open source. In the next chapter, we will be presenting the evaluation of both our framework and the salient results presented here.

Chapter 5

Evaluation

> When you know for yourselves that, "These qualities are skillful; these qualities are blameless; these qualities are praised by the wise; these qualities, when adopted & carried out, lead to welfare & to happiness" - then you should enter & remain in them.
>
> The Buddha[1]

Methods and metrics quantifying scribal behavior were presented in chapter 3 and further exemplified with a case study using Indic script development in chapter 4. Though the case study can be considered to be a partial evaluation of our system, we recognize the need for additional evaluation by expert users in order to understand their views and opinions. Below, we discuss the results and feedback from the user evaluation of our framework (through the prototype) and the salient results of our case study.

5.1 Participants

We recruited 12 domain experts from various research institutions to participate in the evaluation. They were selected based on their expertise in the domain of Indic manuscript/epigraphic studies. Most of them

[1] From *Kālāma Sūtra*

hold academic positions ranging from doctoral/post-doctoral researchers to lecturers and deal with manuscripts on a regular basis. An Indic font designer was recruited to give specific feedback from a typographic point of view, as she is attuned to capture nuances of characters. An informatician working within a manuscript project was recruited to provide feedback from his point of view as a technical user. Thus, the participants were a suitable group performing assessment from different perspectives. Tables 5.1 and 5.2 enumerate the institutions and positions of the participants. Nearly all of them said that their work was *very relevant/absolutely relevant* to the identification/analysis of characters and they performed such analysis quite often. The group was mostly competent with using computers and frequently used them for more than 20 hours per week. Due to scheduling conflicts, one participant was not available to use and evaluate the framework, and therefore was asked to perform only the evaluation of the metrics and the results, after being briefed about the framework.

Institution	Participants
École Française d'Extrême-Orient (EFEO), Pondicherry, India	5
Centre for the Study of Manuscript Cultures (CSMC), University of Hamburg, Germany	3
University of Bologna, Italy	1
University of Tuscia, Italy	1
Independent	1

Table 5.1: Participants' institutions

Position	Participants
Researcher	8
Lecturer	1
Font Designer	1
Informatician	1

Table 5.2: Participants' positions

5.2 Evaluation Design

We recruited participants through our individual research contacts in different institutions. They were mailed individually or asked in-person for their willingness to participate in the study. Part of the evaluation was carried out in-person while visiting the Centre for the Study of Manuscript Cultures (CSMC) at the University of Hamburg to present unrelated work. Others were scheduled through video conferencing (Skype) and asked to access the framework through a remote desktop software (TeamViewer). The evaluation results were recorded using an online questionnaire-based survey (Qualtrics).

Participants were initially given a brief overview of the study and asked to provide non-personal details such as their position, institution and other details to assess their suitability for this study. They were then presented with a manual of the prototype of the framework, which described the functionalities of individual modules and ways to interact with the framework for character analysis. This was typically followed or preceded by a live demonstration of various modules of the framework and their features. They were then asked to use the prototype and experiment with it briefly to familiarize themselves with its interface and operations.

After they felt comfortable with the system, we asked the participants to perform three different tasks on their own. However, they were allowed to ask for assistance and guidance if required. The first task was to perform an end-to-end character analysis through the framework, from importing images to extracting features, for two different characters. To maintain uniformity, we preselected a character set from which participants could choose any two. For the second task, we skipped the importing of image and asked them to select any five distinct characters from the script repository (of the Indic script development dataset) as we wanted them to interact more with other modules. They were suggested to experiment with different kinds of analysis by selecting different trajectories. For the first and second tasks, they were also asked to verify the automatic results, and encouraged to interact and override the system whenever they thought it was appropriate (i.e. when they thought the

results were incorrect and needed to be updated). For the final task, they were asked to select any 10 characters from the Indic development repository and validate the trajectory and stroke segmentation results of the characters (see §4.1).

Once participants' predefined tasks were completed, they were requested to fill the questionnaire regarding the usability, usefulness, user interaction and module-level evaluation of the framework along with the validity of the sampled dataset. To evaluate usability and usefulness, we used standard frameworks that were applicable to our current study (see §5.3.1 & §5.3.2 for more details). We asked them to rate the usefulness specifically in the context of identifying/analyzing characters. For most of the questions, the extent of their agreement to statements was recorded in a five-point Likert scale as follows : *Strongly Disagree*, *Disagree*, *Neither Agree nor Disagree*, *Agree*, *Strongly Agree*. The questionnaire also allowed them to give extensive textual feedback.

After evaluating the framework, participants were presented with the descriptions of the various metrics proposed. The descriptions were simplified, avoiding jargon as much as possible, taking into consideration that the audience was mostly non-technical from the perspective of computer science. After presenting the descriptions, they were asked to rate the metrics in terms of relevancy, usefulness, potential for future/frequent use. Similar to the above, they rated the metrics on a five-point scale with respect to each of these factors. They were then asked to rank the metrics and give specific feedback if any.

Finally, to evaluate the results of our case study we first presented participants with various charts that showed the development of Indic scripts (as seen in figures 4.2, 4.3, 4.4 and 4.5) and briefed them regarding the source and assumptions of the dataset. They were then presented with a subset of results that were obtained by analyzing the visual metrics of the dataset. The extent of their agreement with the results was recorded using a five-point Likert scale as described earlier. We did not perform a complete evaluation of the analyses in our case study as they were too detailed and would require a substantial amount of time to evaluate and check their correctness. Also, for many methods, if we are to accept the underlying data as correct, the results of the methods can be considered

to be valid. Hence, we chose a subset that was easier to validate from the point of view of the participants.

The whole evaluation exercise took about two to three hours to complete. If participants required additional time to evaluate the metrics and results, they were asked to fill in those parts of the questionnaire at a later time convenient to them. The appendices A and B consist of the university ethics approval for the evaluation and the complete questionnaire used by the participants. It is to be noted that the questionnaire was also translated to the Tamil language for participants who were not sufficiently proficient in English.

5.3 Evaluation of the Framework

5.3.1 Usability

The System Usability Scale (SUS) (Brooke et al., 1996) is used to evaluate the usability of our framework. Due to its simplicity, the SUS is one of the most widely used scales for evaluating usability. It consists of 10 questions that evaluate various factors affecting the usability of a system. These questions have a distinct pattern with odd-numbered questions being posed in a positive sense and even-numbered questions in the negative. The Cronbach's alpha score to measure the internal consistency of responses was found to be 0.73 (after adjusting for negative questions), which is fairly good. To calculate the SUS scores, the Likert scale rating is first transposed to a discrete interval ranging from 1 to 5. The score for an individual question is calculated by subtracting 1 from the rating if it is odd numbered or subtracting the rating from 5 if it is even numbered. We can thus notice that the SUS scores are normalized even for negative questions and every question is marked out of a maximum of 4. Table 5.3 lists the median rating and the median SUS score for the individual questions. The individual scores are then summed up to obtain a cumulative score out of 40. This score is then multiplied by 2.5 to get a score out of 100, which is interpreted as the SUS score of the system.

$$\text{SUS}_i = \begin{cases} \text{rating}_i - 1, & i \text{ is } odd \\ 5 - \text{rating}_i, & i \text{ is } even \end{cases}$$

$$\text{SUS} = \sum_i (\text{SUS}_i \times 2.5) \tag{5.1}$$

Figure 5.1 shows the distribution of our SUS scores. The Shapiro-Wilk normality test ($p > 0.05$) shows that the scores can be considered to be normally distributed. The mean of our scores is 79.2 with a standard deviation of 7.5 and a standard error of 2.26. The minimum score obtained is 68.75 and the maximum is 95. Even though it might be tempting to view the total SUS score as a percentage, its interpretation is not very straightforward. We have to apply a percentile based approach to interpret the results. Bangor et al. (2008) performed analysis on numerous SUS surveys and presented results that suggest ways to interpret the SUS scores as a percentile. According to them, the mean score of an SUS survey is 69.69. If the system under evaluation is a GUI, it rises to

Question	Median Rating	Median SUS Score
1. I think that I would like to use this system frequently.	4.5	3.5
2. I found the system unnecessarily complex.	2	3
3. I thought the system was easy to use.	5	4
4. I think that I would need the support of a technical person to be able to use this system.	3	2
5. I found the various functions in this system were well integrated.	5	4
6. I thought there was too much inconsistency in this system.	2	3
7. I would imagine that most people would learn to use this system very quickly.	4	3
8. I found the system very cumbersome to use.	2	3
9. I felt very confident using the system.	4	3
10. I needed to learn a lot of things before I could get going with this system.	2	3

Table 5.3: Individual SUS scores

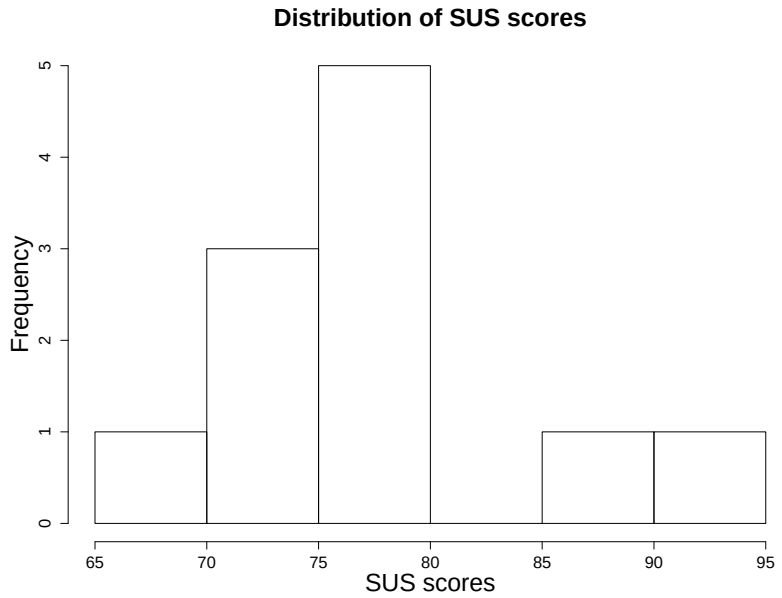


Figure 5.1: Distribution of SUS scores

75.24. Our mean score falls in the third quartile in between the associated adjectives *good* and *excellent* in the interpretation devised by Bangor et al. (2008). The average score is also well above the mean SUS score for a GUI, and our lowest score is approximately the same as the average of a generic SUS score. This suggests that our framework is very usable. Informal feedback during the evaluation in the form of comments shows that participants were finding the system easy to use. Even though they had some trouble understanding the paradigm of the framework, once it had been made clear they found the system quite intuitive to use.

According to Lewis and Sauro (2009), questions 4 and 10 can be combined to a factor called *Learnability*, which is then transformed to a score out of 100.

$$\text{Learnability} = (\text{SUS}_4 + \text{SUS}_{10}) \times 12.5 \tag{5.2}$$

The mean learnability of our framework is found to be 60 with a standard deviation of 11.48 and a standard error of 3.63. The minimum score is 50 and the maximum is 75. The framework appears to score moderately low in terms of learnability. In fact, the lowest individual score is obtained for the fourth question with a median SUS score of 2.

This is to be expected, as none of the participants had any experience using software for paleographic analysis, and furthermore, our framework operates on an unconventional paradigm for analysis. However, most of them found the system to be easy to use and well integrated (median SUS score: 4) and would use the system frequently (median SUS score: 3.5). It shows that users face a significant learning curve to use our framework. This is corroborated by the verbal feedback that was received during the evaluation. The participants who found the system initially difficult to use, commented that the system would be easier to use after some practice.

In terms of user interaction with the system, participants found the guiding factor behind the framework interesting. One participant was particularly enthusiastic regarding the creation of spline representations and a few others appreciated the ability of the framework to reconstruct trajectories. We observed that participants were quite engaged, interacting with the framework and overriding its results when needed. They appeared to appreciate the fact that they could improve the results (under the guiding paradigm) and hence contribute to the framework's results. Some had specific issues in understanding the *segmentation* module and the underlying principle had to be reiterated. We understand that if the system is to be adopted widely, the underlying paradigm of the framework (and the framework itself) must be communicated in a very accessible manner.

The following statements were made regarding usability when they were asked to provide qualitative feedback about the positive aspects of the framework.

> "Surprisingly easy to use, and effective"

> "easy to use, good output of character forms"

> "The interface was well planned. Can automate a lot of useful statistical analysis that would be time consuming to arrive at otherwise"

> "Relatively easy to use; it could be use [sic] to determine whether different manuscripts have been written by the same hand or not"

When also asked to enumerate the negative aspects of the system:

> " due tue [sic] my inexperienced [sic] with numbers and statistic [sic] I would find some difficulties in comparing numbers."

> "I don't see many negative aspects of this system. Researchers might have initial difficulties understanding all of the data or knowing what do with it, but I think it could answer a number of questions. There are, however, some limitations. For the input of the strokes it is necessary to know how the script was written which may have varied over time and without this information we cannot accurately represent what we find in our manuscripts."

The feedback is very positive. However, a participant mentioned their concern about the accuracy of reconstructed trajectories (in general). There were also some features expected out-of-the-box from the system. For instance, manuscript experts are generally uncomfortable dealing with quantitative analysis. It was suggested that it would be very helpful if the system came with in-built statistical techniques apart from allowing users to perform open-ended analyses using the metrics.

5.3.2 Usefulness

For evaluating the usefulness of the framework, we used the *Perceived Usability* metric by F. D. Davis (1989). It consists of six questions that can be answered with a Likert scale. The Cronbach's alpha for this part of the questionnaire is found to be 0.92 and shows that the data has very high internal consistency. Table 5.4 shows the median and minimum ratings for each question. The mean usefulness rating was found to be 4.13 with a standard deviation of 0.72 and a standard error of 0.21.

The minimum rating is 2.83 $\approx$ 3, which is more or less a neutral rating. This demonstrates that participants have found the framework to be quite useful in their field. Figure 5.2 shows the distribution of the mean usefulness ratings across participants.

Questions	Median. Rating	Min. Rating
1. Using the system in my job would enable me to accomplish tasks more quickly	5	3
2. Using the system would improve my job performance	4	3
3. Using the system in my job would increase my productivity	4	2
4. Using the system would enhance my effectiveness on the job	4	2
5. Using the system would make it easier to do my job	4	2
6. I would find the system useful in my job	4	4

Table 5.4: Median and minimum ratings for the usefulness questionnaire

However, based on informal feedback, a few participants were not particularly convinced about the use of the system directly to their line of work, even though they rated the system favorably in the (prospective) context of analyzing characters. One of the reasons is that these participants were more involved in transcriptions of manuscripts in their main work and not directly related to studying character shapes (to identify scribes etc). But verbal feedback from those involved in recognizing hands commented that the framework might be very useful in their line of work. A few participants showed interest in using the framework for their personal research to analyze their datasets. Even though the framework is proposed in the context of descriptive paleography, the participants appear to be more inclined towards detection of hands. We posit that this is probably due to their own familiarity and relevance of their work to hands detection. But as mentioned in sections 1.1 and 3.2.5, the framework does provide better interpretable metrics and methods for

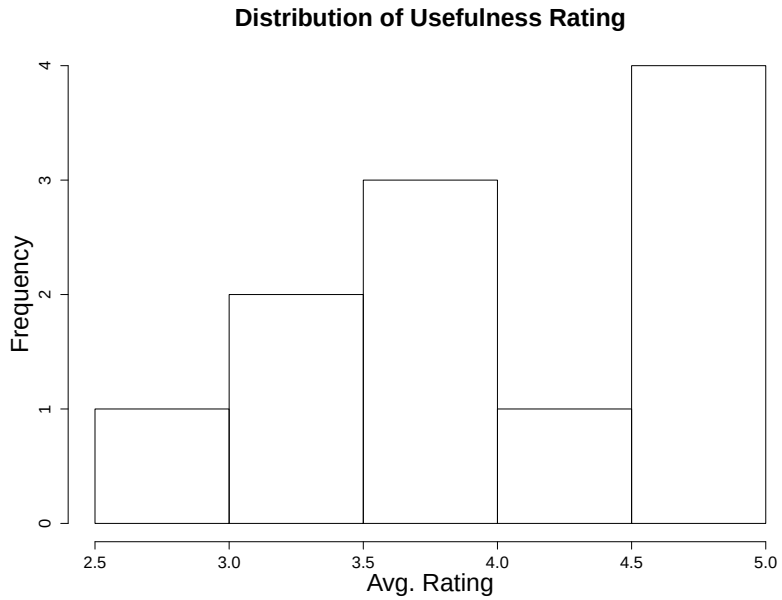


Figure 5.2: Distribution of usefulness

such applications. One particular informal feedback stated that "it allows them to perform innovative analysis that usually will not be performed manually". For instance, it would be possible to "group characters based on the conditions of writing" and that it will "allow them to support their qualitative findings quantitatively". It appears that some participants do appreciate the innovative analysis that the framework would enable them to perform. Also, it was noted that the "metrics probably need usable documentation before experts can make proper use of them".

Similar to the previous section, we report below the qualitative feedback regarding the usefulness of the framework. Though it largely echoes the informal feedback that was received, at least one participant was inclined to use it to study the "evolution of writing systems" and appreciates the framework's use with descriptive paleography.

> "It produces an objective description of the characters "

> "This system provides a relatively easy way to store data concerning how a character is written, including number of strokes, direction of strokes, and various other parameters like velocity and trajectories. I could imagine a number of

> questions regarding the evolution of writing systems and the analysis of different hands. It was also relativelly [sic] easy to learn to use."

> "The system can help to understand if the hand who wrote the manuscript was the same or not. The system can compare same script from the same manuscript. / The system can quantitatively evaluate what I can qualitaltively [sic] already evaluate."

> "In my opinion, the system would prove really effective for the palaeographic study of Indian scripts, as well as in determining "styles" of writing which could turn out to be typical of one specific scribe or scriptorium or regional area or period"

> "The system is a highly promising tool for the study of scripts. I would be eager to test it on sets of Grantha characters taken from selected manuscripts of known date and provenance. In order to determine whether the system can identify the ductus of a specific scribe, it would be perhaps profitable to test it with sets of characters taken from manuscripts written by the same scribe as well as by different scribes having similar ductus"

5.3.3 Framework Modules

We asked participants to provide feedback regarding the individual modules of the framework, namely spline conversion, trajectory reconstruction and stroke segmentation. For each module, they were asked about relevancy, effectiveness and the extent of agreement with the results of automatic analysis using the below set of statements.

1. I find this module very relevant to the workflow
2. I find the module very effective
3. I agree with the automatic analysis results of the module

The Cronbach's alpha value is 0.91 for this questionnaire, which demonstrates very high internal consistency. Tables 5.5, 5.6, and 5.7 show the responses from them. Note that the five-point Likert scale is reduced to a three-point scale to simplify the reporting of agreements. It can be seen that nearly all of the participants find the modules relevant and effective and most also agree with the results of the automatic analysis.

	Agree	Neither Agree or Disagree	Disagree
Relevancy	10	1	0
Effectiveness	10	1	0
Automatic Analysis	11	0	0

Table 5.5: Evaluation of spline conversion

	Agree	Neither Agree or Disagree	Disagree
Relevancy	9	2	0
Effectiveness	11	0	0
Automatic Analysis	9	2	0

Table 5.6: Evaluation of trajectory reconstruction

	Agree	Neither Agree or Disagree	Disgree
Relevancy	9	1	1
Effectiveness	10	0	1
Automatic Analysis	10	1	0

Table 5.7: Evaluation of segmentation

The following qualitative feedback was received with respect to the trajectory reconstruction module.

> "At times it is quite difficult to decide which of two or more alternative trajectories is the one which was used actually by the scribes".

> "The trajectory reconstruction is usually accurate or provides enough options to select the proper trajectory. In any case, it is interesting to have to think about the proper trajectory or see how actual practice differs from what the program guesses".

The comments are understandable as it is up to the user to select a relevant trajectory. But it is interesting to note that it also encourages people to think about unconventional trajectories.

Additionally, participants were asked to rate their involvement in the framework. Table 5.8 shows the median responses. It is evident that most of the users find the framework intuitive and the ability to intervene very useful. The framework allows them to explore while providing a great deal of control at the same time. Thus, it can be seen that the framework encourages open-ended exploration and analysis, which we think is crucial. This was also mentioned by one of the participants as an informal comment.

Question	Median Rating
1. I found the workflow of the system intuitive	4.5
2. I found the ability to manually intervene/override very useful	5
3. I felt the system allowed me to explore different kinds of analysis	4
4. I felt the system allowed me a great deal of control with analysis	4

Table 5.8: User interaction with framework

When asked if participants wanted any changes to the workflow of the framework, all but one said they did not want any changes. The lone participant who had suggested changes, requested an additional feature to display multiple glyphs side by side for analysis. This amounts to

a slight change in presenting the information when analyzing multiple characters. Overall, it is inferred that the participants are content with the current workflow of the framework.

5.3.4 Evaluation of the Case Study Dataset

All the analyses presented as a part of our case study depend on the data present in the script repository (for Indic script development). Therefore, we asked participants to evaluate the results of character analyses in the dataset by validating the results of the trajectory reconstructions and segmentations, and then record their agreements. It is a daunting task to verify all the 700 odd characters in the repository. Instead, each participant was asked to verify any 10 characters in the dataset (some chose to check more characters). In this way, around 120 characters were verified by the participants, which amounts to approximately 15% of the total dataset. This is a significant sampling and hence can be considered as representative of the entire dataset. From table 5.9, it is very clear that participants agree with our character analyses with near unanimity. The one participant who did not agree is neutral to the validity of character analysis results in the dataset.

	Agree	Neither Agree or Disagree	Disagree
Trajectory Reconstruction	10	1	0
Stroke Segmentation	10	1	0

Table 5.9: Evaluation of data in script repository

5.4 Evaluation of Metrics

Table 5.10 displays the median ratings for individual metrics under different criteria - relevancy, usefulness, (potential for) future use, (potential for) frequent use. Figure 5.3 shows the range of the corresponding individual responses. On the basis of quantitative evaluation, it can be

clearly seen that all of our metrics have near unanimous high ratings (4 and above) in every criterion. This affirms that most experts agree that our metrics are useful and relevant. However, we focus on metrics that received comparatively lower ratings. It can be seen that kinematic metrics consistently show lower ratings compared to visual metrics. This suggests that the participants were thinking in terms of conventional paleographic analysis and were not attuned to descriptive analysis. Nevertheless, they do see these metrics as very important. Interestingly, *overall length* scores low among visual metrics as compared to individual stroke lengths, which has a higher rating. We suppose this is because the former was seen as redundant in the light of the latter.

Participants were also asked to rank the metrics based on the order of importance. Out of the eleven participants, only four ranked the metrics based on their order of priority. It must be said that some participants had a subjective view what *importance* meant. One participant noted "I've organised according to the information I would look for in the order of what I'd want to see. Many of them are equally important for my understanding of what's happening with a script". In any case, the order may be taken as indirectly pointing out to the hierarchy of importance of the metrics. From table 5.11, which enumerates the ranking of the metrics, we can observe that most participants (Ranking 1 to 3) think that visual metrics are more important than the others. Only one ranking had more kinematic metrics in the top list. This is similar to what we had seen in the individual ratings of the metrics, namely that the participants think in terms of conventional analysis. In detail, however, the ranking appears to be very divergent with no common features in the top five that are common in all the four lists. (Paucity of responses is also partially to blame for this.)

Regarding the application of the metrics, as seen above and in section 5.3.2 about usefulness, most of the participants are more interested in applying the metrics in the identification of hands. However, interestingly few participants have particularly noted some non-traditional analyses that could be performed using the metrics. The qualitative feedback obtained are given below.

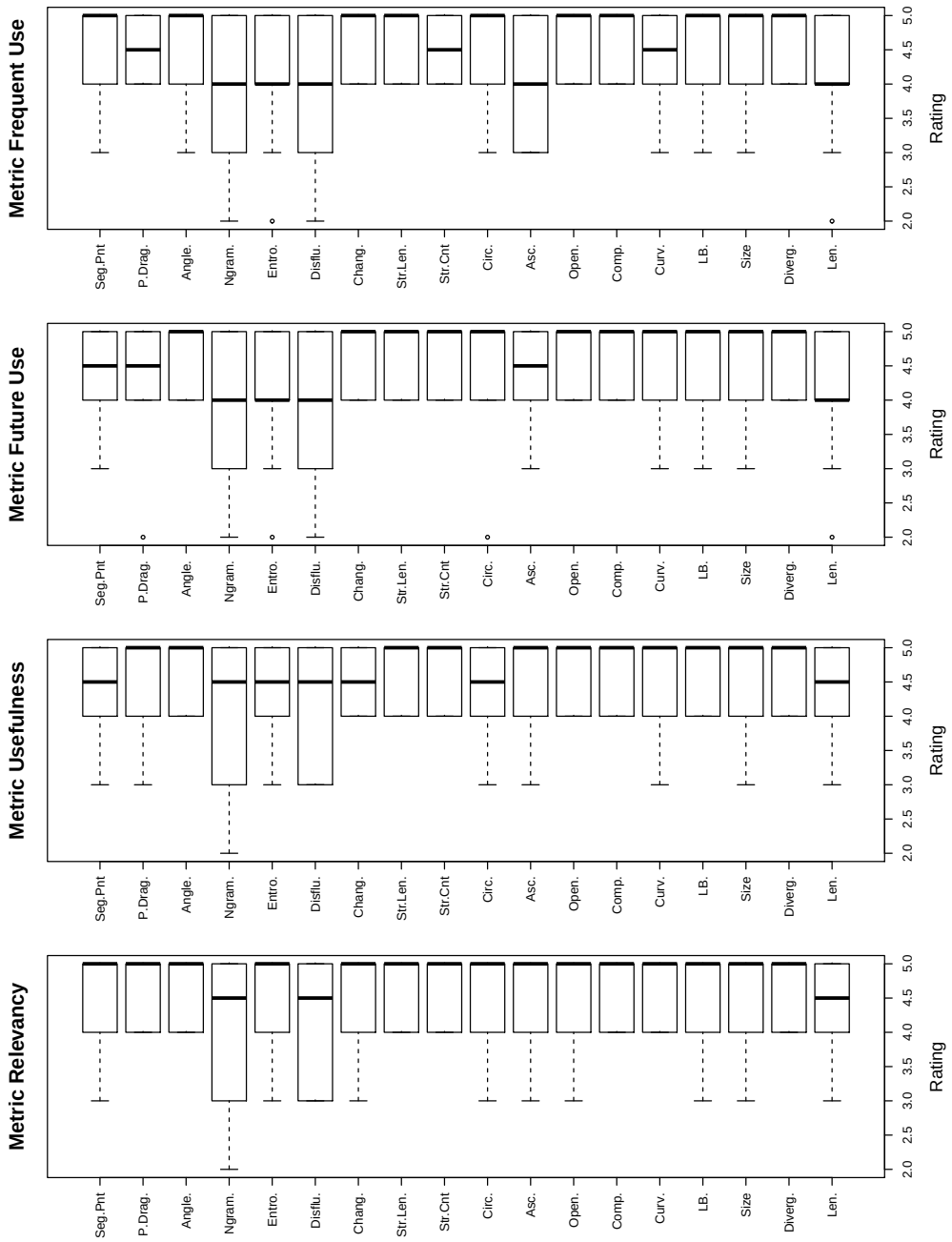


Figure 5.3: IQR for metrics under various evaluation factors

Metric	Relevancy	Usefulness	Future Use	Frequent Use
Length	4.5	4.5	4	5
Divergence	5	5	5	5
Size	5	5	5	5
Length-Breadth Index	5	5	5	5
Average Curvature	5	5	5	4.5
Compactness	5	5	5	5
Openness	5	5	5	5
Ascendance & Descendance	5	5	4.5	4
Circularity & Rectangularity	5	4.5	5	5
Stroke Counts	5	5	5	4.5
Stroke Length	5	5	5	5
Changeability	5	4.5	5	5
Disfluency	4.5	4.5	4	4
Entropy	5	4.5	4	4
N-Gram model of scripts	4.5	4.5	4	4
Angle-based metrics	5	5	5	5
Pen-Drag distance	5	5	4.5	4.5
Number of Segmentation points	5	4.5	4.5	5

Table 5.10: Median ratings for individual metrics under different criteria

"Recognizing scribal hands"

"I would apply the metrics to the study of the Grantha script in manuscripts. I would try to quantitatively define the ductus of a scribe, the different writing styles which are characteristic of different geographic areas and, possibly, the development of single characters in specific areas."

"There are a few metrics I'd want in the database like 'length' and 'No. of Landmark Points' that would be useful to input

	Ranking 1	*Ranking 2*	*Ranking 3*	*Ranking 4*
1	Ascendence and Descendance (V)	Length (V)	Landmark Points (K)	Openness (V)
2	Circularity and Rectangularity (V)	Divergence (V)	N-gram Model (K)	Ascendence and Descendance (V)
3	Avg. Curvature (V)	Size (V)	Size (V)	N-gram Model (K)
4	Changeability (K)	LB Index (V)	LB Index (V)	Circicularity and Rectangularity (V)
5	LB Index (V)	Avg. Curvature (V)	Ascendence and Descendance (V)	Stroke Counts (K)
6	Compactness (V)	Changeability (K)	Avg. Curvature (V)	Stroke Lengths (K)
7	Disfluency (K)	Ascendence and Descendance (V)	Openness (V)	Changeability (K)
8	Entropy (K)	Circularity and Rectangularity (V)	Changeability (K)	Disfluency (K)
9	Length (V)	Compactness (V)	Compactness (V)	Entropy (K)
10	Stroke Counts (K)	Disfluency (K)	Circularity and Rectangularity (V)	Compactness (V)
11	Stroke Lengths (K)	Entropy (K)	Stroke Counts (K)	Divergence (V)
12	Divergence (V)	N-gram Model (K)	Stroke Angles (K)	LB Index (V)
13	N-gram Model (K)	Stroke Angles (K)	Divergence (V)	Pen-Drag (K)
14	Stroke Angles (K)	Pen-Drag (K)	Disfluency (K)	Avg. Curvature (V)
15	Pen-Drag (K)	Landmark Points (K)	Length (V)	Landmark Points (K)
16	Landmark Points (K)	Openness (V)	Entropy (K)	Length (V)
17	Openness (V)	Stroke Lengths (K)	Stroke Lengths (K)	Size (V)
18	Size (V)	Stroke Counts (K)	Pen-Drag (K)	Stroke Angles (K)

Table 5.11: Ranking of metrics. Visual metrics are marked with *V* and kinematic metrics with *K*

> in order to derive further analysis in several places by default. I wouldn't use them actively. Cognitive metrics to arrive at a hypothetical underlying skeleton and the N-Gram model to predict what might be the missing characters' structure would be extremely useful. If this N-Gram model can be linked with the language behaviour, once again, it might be possible to predict the character itself and its possible shape. This is often the most challenging bit and having automated suggestions would be invaluable".
>
> "I would be most interested in determining the number of characters per line (and hence per page etc.). I would also be interested in seeing whether it's possible to identify specific scribes based on quantifiable data as apposed to simply intuition. Another application would be to see how characters change when written in a more hurried or casual manner".

5.5 Evaluation of Quantitative Analysis

We evaluated the results of our case-study starting with the trends that were shown in section 4.2.1. To briefly recall, we analyzed the metrics from various scripts and noticed specific trends during the development of scripts. We summarized the reported trends to eight distinct statements and asked participants to record the extent of agreement with the statements on a five-point Likert scale. Table 5.12 reports the corresponding median ratings, and figure 5.4 shows the interquartile range (IQR) for the ratings. It must be noted that out of the total eleven participants who had performed the evaluation study, only ten responded to this part of the questionnaire.

It can be observed that there are some disagreements regarding the trends of *rectangularity* and *compactness*. Rectangularity is a different measure from a character's *geometricity*. Rectangularity of a circle will be comparatively low but still considerably higher than that of a non-geometric figure. For this reason, even rounded scripts like Grantha and Kannada exhibit comparatively high rectangularity. This probably

Statement	Tamil	Grantha	Kannada	Devanagari
1. The length of characters has increased over time	4	4	4.5	4
2. The length-breadth index of the characters has increased over time	4	4	4	4
3. The circularity of the characters has increased over time	4	4	4	4
4. The rectangularity of the characters has increased over time	4	3.5	2.5	4
5. The openness of the characters has decreased over time	4	4	4	4
6. The compactness of the characters has decreased over time	3	3	3	3
7. The average curvature of the characters has increased over time	4	4	4.5	4
8. The characters have become more complex to write over time	4	4.5	4.5	5

Table 5.12: Median ratings for feature trends

indicates that only one of the measures should be used to indicate geometricity of a character. Also, we think that the compactness values are a bit complicated to judge visually (given the short span of the survey). Hence, the median rating for that particular feature is in the range of neutrality. Apart from those two metrics, participants agree with other features' trends, which shows that these metrics are highly correlated with human judgments as well. Figure 5.4 shows few outliers (as opposed to the corresponding median ratings), out of which very few disagree with our statements and most others either strongly agree or are neutral. This again indicates a high level of agreement with our analysis of trends.

We then questioned participants regarding their agreement with various factors that are proposed to have caused diversification of scripts as seen in section 4.2.2. Similar to the evaluation of trends, this was summarized into affirmative statements to quantify their agreements on a Likert scale. Table 5.13 summarizes the responses through their median values and figure 5.5 shows the variations in the responses through their

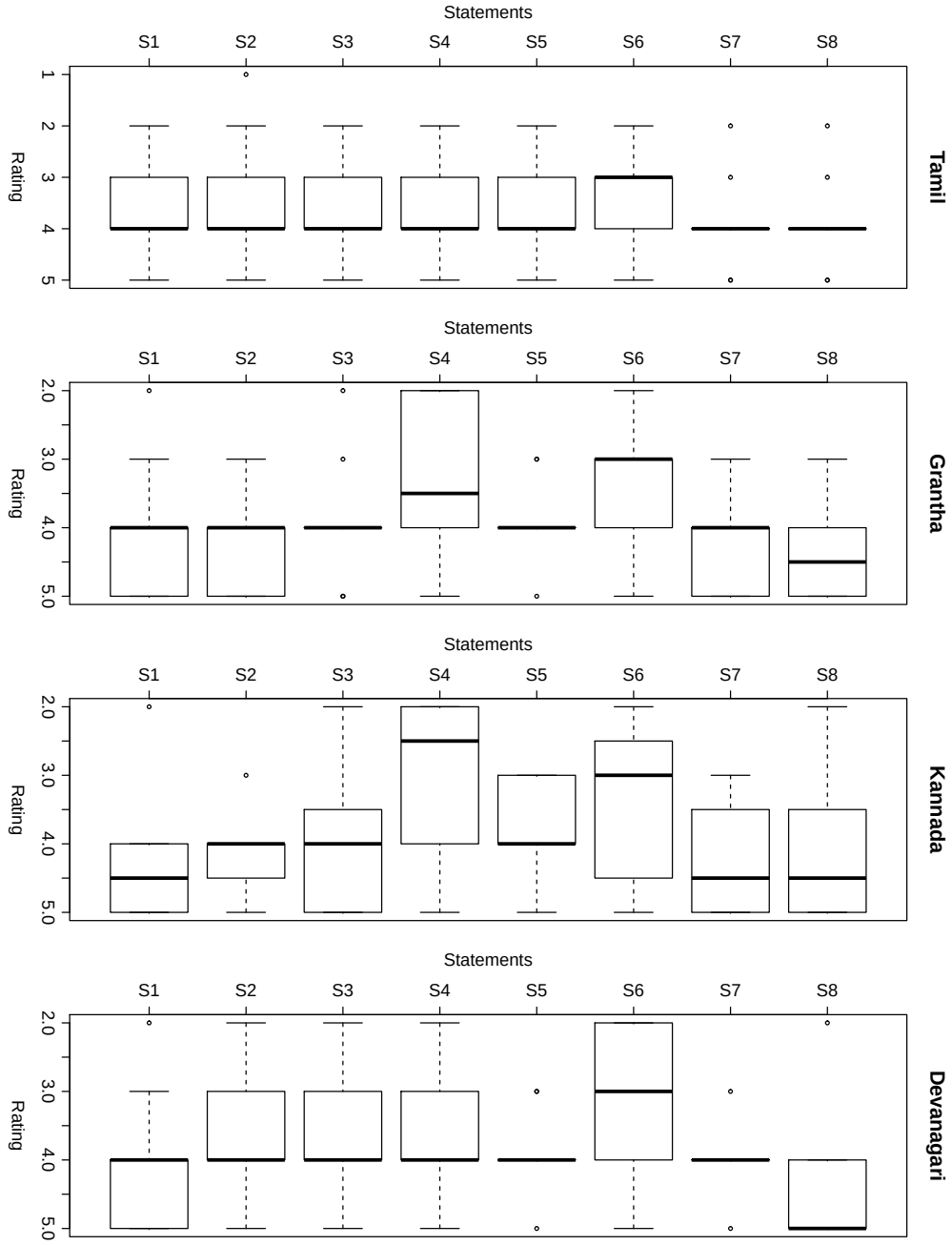


Figure 5.4: Interquartile range of responses for metrics' trends

interquartile range. The participants appear to have essentially agreed with our diversification factors with a median range of 4 (*Agree*) for all the statements. Similar to that of the evaluation of trend, there are some variations in the individual responses. But except for one or two of these responses, overall they appear to be in agreement (or neutral).

Statements	Median Rating
1. The characters diversified based on Compactness	4
2. The characters diversified based on Average Curvature	4
3. The characters diversified based on Circularity	4
4. The characters diversified based on Rectangularity	4

Table 5.13: Median ratings for script diversification

The previous exercise was repeated but within the context of spread of variations among Indic scripts (see section 4.2.3). Table 5.14 and figure 5.6 describe the responses from participants. It is evident that again participants mostly agree with us.

Statements	Median Rating
1. Brahmi characters are very similar to each other compared to other characters	4
2. The characters gained more symmetry, length and less openness [in the intermediate stage]	4
3. The characters at the end stage gained more symmetry and curvature [in their final form]	4
4. The characters at the end stage gained more length and less openness [in their final form]	4
5. I see a specific pattern in the way characters have evolved	4

Table 5.14: Median ratings for spread of variations

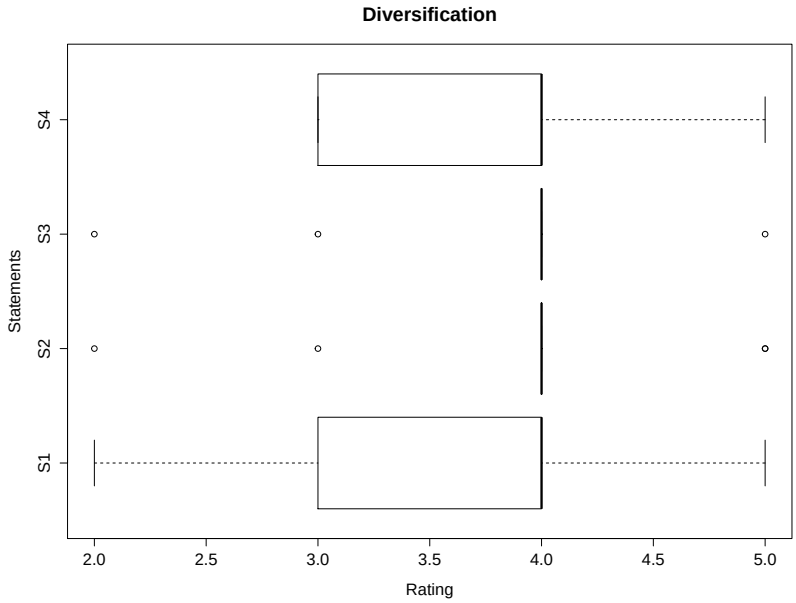


Figure 5.5: Interquartile range of responses for script diversification

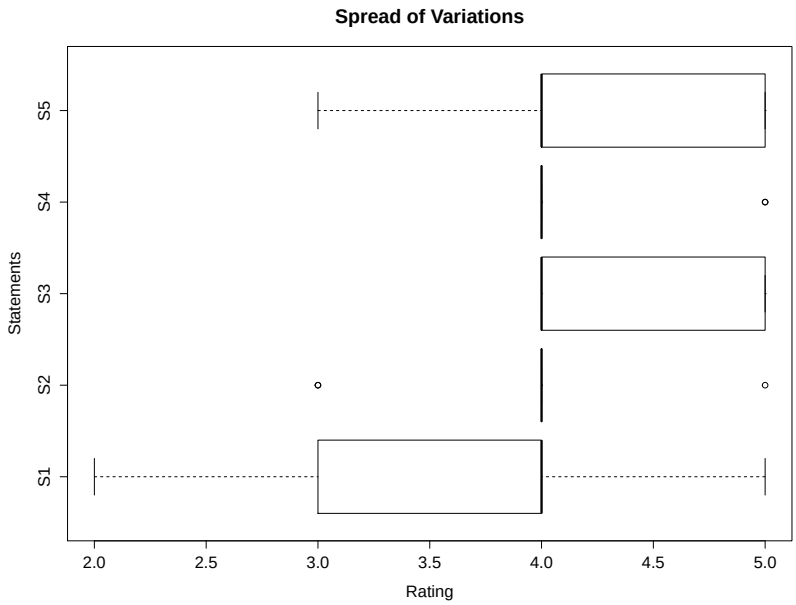


Figure 5.6: Interquartile range of responses for spread of variations

With reference to the distinctiveness of scripts discussed in section 4.2.6, we asked participants if the characters within scripts have in general become similar over time. Out of the total ten participants, eight agree with the results, only one disagrees and the remaining one is neutral. The disagreeing participant had a background in philology. As a partially subjective measure this minor variation is to be expected but overwhelmingly participants do agree with our analysis.

All participants unanimously agreed that they found quantitative analyses like these very interesting. Seven out of the total ten respondents said they were interested in performing such descriptive analysis in the future, one said they would not and the remaining two responded neutrally. To recall the discussion regarding usefulness and metrics (see sections 5.3.2 and 5.4), most participants viewed the framework and the metrics in the context of only hands detection and only a few foresaw applications in a descriptive context. However, after being exposed to our analysis most participants appeared to be open to descriptive analysis, assuming the required toolsets to perform such analyses existed. This is very encouraging. We think our proposed framework equips experts with a well defined functional toolset to perform such innovative analyses.

5.6 Summary

We presented the user study that was performed to evaluate our proposed system and the salient results of our case study. First, the detailed design of the evaluation study was explained along with an overview of participants. We then discussed the usability and usefulness of the framework as judged by the participants, followed by an evaluation of the individual modules of the framework and the script repository of the case study. It was shown that the participants overwhelmingly had positive and encouraging opinions about the framework. We then proceeded to discuss the evaluation of the metrics, followed by a detailed description of the evaluation of the quantitative results of our case-study. Though there were some disagreements, participants agreed with our metrics and with the results of our quantitative analyses on the whole. They

also found our descriptive analyses to be interesting and might perform such analysis in the future. In the next chapter, we will present our final conclusions and discuss future work.

Chapter 6

Conclusion and Future Work

> The Tathāgata (Buddha) reveals his teachings necessarily by means of expressive signs (*monji*).
>
> Kōbō-Daishi Kūkai[1]

This thesis began as an ardent attempt to quantify *the art of a scribe* as Stansbury (2009) put it. The basic premise of our research as exemplified by the central hypothesis (see §1.2) was to investigate whether handwriting information can aid digital paleographic analyses by providing insights into scribal behavior through quantitative methods. As discussed in section 1.1, we sought to validate this with three research questions that defined the scope of our work. Below, these are enumerated again. Through our work we have been able to answer these questions within the reasonable limitations and assumptions (see §6.1) that were set forth by us.

1. What is the computational framework required to extract handwriting information present in characters and analyze them?
2. What metrics and methods are required to perform quantitative analysis of scribal behavior using the extracted information?
3. How can the recovered information be directly used to quantitatively study changes in scribal behavior?

[1] From The Meaning of Sound, Word and Reality (*Shōji Jissōgi*)

We began by studying various existing digital paleographic systems and identifying their shortcomings to quantify scribal behavior. We proceeded by surveying various feature sets proposed for characters within quantitative paleography and also domains outside of it such as gesture recognition. We also ventured into handwriting modeling techniques as a means of studying changes in scribal behavior.

We proposed a novel computational framework that operates on the paradigm of handwriting information of characters, i.e. Ductus, to quantitatively study scribal behavior. The framework recovers trajectories of characters, which are used as the basis to perform quantitative analyses. With this recovered information, we suggested a method to decompose characters into their underlying primitives, i.e. *strokes*, using handwriting kinematics, and to construct a hierarchical structure. We then described a range of intuitive metrics to quantify scribal behavior, which were to be extracted from the aforementioned stroke structure of characters. We also proposed the use of the Sigma-Lognormal model of handwriting production to be used in conjunction with our framework to quantitatively study changes in scribal behavior in finer detail.

Using the development of Indic scripts as a case study, we further illustrated the effectiveness of our framework and the various applications of the proposed metrics to discover interesting trends and phenomena in scribal behavior during the development. We also showcased a number of statistical methods that can be used with our metrics to perform descriptive analysis. Using the same case study, we demonstrated the application of the Sigma-Lognormal model (along with the framework) by modeling changes in handwriting to quantitatively analyze the divergence of Brahmi into early Tamil and early Kannada.

We then performed an elaborate user study to evaluate our framework and the salient results of our case study. The framework was studied mainly in the context of its usability and usefulness, and the case study with respect to the extent of user agreement with its results. We showed that domain experts found the framework and metrics interesting and useful, and also agreed with most of the results of our analysis. We also found a strong indication that our research (and case study) encouraged experts to delve into descriptive paleography.

To summarize, the following are provided as answers to our earlier research questions and subsequently as original contributions to knowledge (also see §1.4).

1. A framework (see §3.2) that recovers handwriting information of characters, and uses that information coupled with handwriting kinematics to decompose characters into strokes and construct a hierarchical stroke structure.

2. A number of intuitive metrics (see §3.2.5) that quantify handwriting behavior derived using the recovered information and structure, which coupled with proper statistical methods can be used to perform quantitative analysis of scribal behavior.

3. Using recovered trajectory (from our framework), handwriting modeling techniques such as the Sigma-Lognormal model (see §4.3) can be directly used to mathematically model changes in scribal behavior, which can then be used to quantitatively study them.

We have additionally released the source code of the prototype implementation of our framework and the data files of our case study as additional contributions to the wider research community.

Going back to our research hypothesis, these answers along with our case study and evaluation clearly validate our original hypothesis. Handwriting information can certainly provide great insights into scribal behavior and assist in creating intuitive computational methods for quantitative digital paleographic analyses. Thus, *the art of a scribe* can indeed be quantified through appropriate methods and then be studied.

6.1 Limitations

Apart from the overall limitation set out in section 1.5, we detail below the specific limitations of our work. As explained earlier in section 3.2, our approach assumed an ideal motion of writing implements on surfaces. Therefore, it can only be applied to writing styles and materials that involve free hand movements. The proposed methods and metrics must

be applied based on this idealized assumption, which may limit the interpretation of the obtained results. This assumption frequently does not hold completely true for paleographic scripts. Our framework will need to be adapted for different modes of writing that realistically reflect the actual process of writing under consideration.

With regards to our metrics, we propose only a relevant subset. However, using the stroke structure a larger number of intuitive metrics can be obtained. Also, the metrics are more focused on unistroke characters than multi-stroke characters. The framework is adaptable and open-ended, which makes it possible to accommodate these changes to overcome the limitations with relative ease.

6.2 Future Work

The main driving force behind our research was to support experts performing descriptive paleography by equipping them with effective computational methods and metrics. As explicitly reiterated several times during our expert evaluation, one of the major applications of our research would be in the detection of hands. But, given our focus on descriptive analyses, we did not deviate our efforts to delve into the problem of classifying glyphs. However, it is essential that our approach is tried out in the context of hands detection to evaluate its applicability and overall fit. The amount of assistance our approach could provide experts at least with manual or preliminary classification will be an interesting problem to investigate. In fact, during our evaluation, a researcher from the University of Bologna became interested in employing our approach to detect hands in Grantha manuscripts. The researcher is involved in the chronological (and geographical) development of the Grantha script based on palm leaf manuscripts starting from the early 18th century. Future work in this direction may yield further insights regarding the effectiveness of our work in this context.

In terms of handwriting modeling, as noted in section 4.3, the shape-changing behavior was modeled manually through intuitive heuristics. An automated approach would require a cost function that is optimal

for black-box optimization methods. Along similar lines, formulating heuristics that properly guide the process of shape-change, in spite of the huge solution space, is required as well. With a well defined and customized heuristics, it would also be possible to automatically generate putative intermediate forms. Moreover, using the transformation matrices to reconstruct missing characters in a given set is an interesting application requiring further work.

Our research is the first step in the direction of exploring a quantitative approach for descriptive paleography. We hope more methods and metrics will follow, overcoming the limitations and expanding our work even further.

Appendix A

University Ethics Approval for User Evaluation Study

University of St Andrews
Scotland's first university – 1413

University Teaching and Research Ethics Committee Sub-committee

30th November 2015
Vinodh Rajan Sampath
School of Computer Science

Ethics Reference No: *Please quote this ref on all correspondence*	CS11844
Project Title:	Evaluation a Script Analysis Framework in the context of Digital Paleography
Researchers Name(s):	Vinodh Rajan Sampath
Supervisor(s):	Dr Mark-Jan Nederhof

Thank you for submitting your application which was considered at the Computer Science School Ethics Committee meeting on the 24th November 2015. The following documents were reviewed:

1. Ethical Application Form 24/11/2015
2. Participant Information Sheet 24/11/2015
3. Consent Form 24/11/2015
4. Debriefing Form 24/11/2015
5. Advertisement 24/11/2015

The University Teaching and Research Ethics Committee (UTREC) approve this study from an ethical point of view. Please note that where approval is given by a School Ethics Committee that committee is part of UTREC and is delegated to act for UTREC.

Approval is given for three years. Projects, which have not commenced within two years of original approval, must be re-submitted to your School Ethics Committee.

You must inform your School Ethics Committee when the research has been completed. If you are unable to complete your research within the 3 three year validation period, you will be required to write to your School Ethics Committee and to UTREC (where approval was given by UTREC) to request an extension or you will need to re-apply.

Any serious adverse events or significant change which occurs in connection with this study and/or which may alter its ethical consideration must be reported immediately to the School Ethics Committee, and an Ethical Amendment Form submitted where appropriate.

Approval is given on the understanding that the 'Guidelines for Ethical Research Practice' https://www.st-andrews.ac.uk/utrec/guidelines/ are adhered to.

Yours sincerely

pp. Robertson

Convenor of the School Ethics Committee

Ccs Supervisor
School Ethics Committee

ethics-cs@st-andrews.ac.uk

The University of St Andrews is a charity registered in Scotland: No SC013532

Appendix B

Questionnaire for User Evaluation Study

University of St Andrews

English ▾

Default Block

P1. Participant Information

Project Title
Evaluation of a Script Analysis Framework in the context of Digital Palaeography

What is the study about?

We invite you to participate in a research project about a Script Evaluation Framework used in the context of digital palaeography. The framework allows you to analyse characters using handwriting information to derive script-independent metrics. We'll be evaluating the usability of the framework, the metrics and also some results derived using the framework.

This study is being conducted as part of my, Vinodh Rajan Sampath's PhD Thesis in the School of Computer Science.

Do I have to take Part?

This information sheet has been written to help you decide if you would like to take part. It is up to you and you alone whether or not to take part. If you do decide to take part you will be free to withdraw at any time without providing a reason.

What would I be required to do?

You will be asked to evaluate a Script Analysis Framework via a remote desktop software. This would consist of performing a set of tasks using the framework. Then you will be asked to fill in a questionnaire to evaluate this framework. This would be followed by another set of questionnaire to evaluate metrics used in the framework and also results obtained using the framework.

Following, there would be a short free-form interview to have a summary discussion.

The whole exercise would take around 2 to 3 hours.

Will my participation be Anonymous and Confidential?

Only the researcher(s) and supervisor(s) will have access to the data, which will be kept strictly confidential. Your permission maybe sought in the Participant Consent form for the data you provide, which will be anonymised, to be used for future scholarly purposes.

Storage and Destruction of Data Collected

The data we collect will be accessible by the researcher(s) and supervisor(s) involved in this study only, unless explicit consent for wider access is given by means of the consent form. Your data will be stored for an indefinite period in an anonymised format on a computer system.

What will happen to the results of the research study?

The results will be finalised by April 2016 and will be published in the form of a research paper or note, and/or written up as part of my PhD Thesis

Questions

You will have the opportunity to ask any questions in relation to this project before giving completing a Consent Form.

Consent and Approval

This research proposal has been scrutinised and been granted Ethical Approval through the University ethical approval process.

What should I do if I have concerns about this study?

A full outline of the procedures governed by the University Teaching and Research Ethical Committee is available at http://www.st-andrews.ac.uk/utrec/guidelinespolicies/complaints/

Contact Details

Researcher: Vinodh Rajan Sampath
Contact Details: vrs3@st-andrews.ac.uk

Supervisor: Dr Mark-Jan Nederhof
Contact Details: mn31@st-andrews.ac.uk

Participant Consent Form

Coded Data

Project Title
Evaluating a Script Analysis Framework in the context of Digital Palaeography

Researcher(s) Name(s)
Vinodh Rajan Sampath
vrs3@st-andrews.ac.uk

Supervisors Names
Dr Mark-Jan Nederhof
mn31@st-andrews.ac.uk

The University of St Andrews attaches high priority to the ethical conduct of research. We therefore ask you to consider the following points before signing this form. Your signature confirms that you are happy to participate in the study.

What is Coded Data?
The term 'Coded Data' refers to when data collected by the researcher is identifiable as belonging to a particular participant but is kept with personal identifiers removed. The researcher(s) retain a 'key' to the coded data which allows individual participants to be re-connected with their data at a later date. The un-coded data is kept confidential to the researcher(s) (and Supervisors). If consent it given to archive data (see consent section of form) the participant may be contacted in the future by the original researcher(s) or other researcher(s).

Consent
The purpose of this form is to ensure that you are willing to take part in this study and to let you understand what it entails. Signing this form does not commit you to anything you do not wish to do and you are free to withdraw at any stage. Material gathered during this research will be coded and kept confidentially by the researcher with only the researcher and supervisor having access. It will be securely stored for an indefinite period.

Please answer each statement concerning the collection and use of the research data.

P3. I have read and understood the information sheet.

- Yes
- No

P4. I have been given the opportunity to ask questions about the study.

- Yes
- No

P5. I have had my questions answered satisfactorily.

- Yes
- No

P6. I understand that I can withdraw from the study at any time without having to give an explanation.

- Yes
- No

P7. I understand that my data will be confidential and that it will contain identifiable personal data but that will be stored with personal identifiers removed by the researcher and that only the researcher/supervisor will be able to decode this information as and when necessary.

- Yes
- No

P8. I understand that my data will be stored for an indefinite period.

- Yes
- No

P9. I have been made fully aware of the potential risks associated with this research and am satisfied with the information provided.

- Yes
- No

P10. I agree to take part in the study.

- Yes
- No

P11.

Participation in this research is completely voluntary and your consent is required before you can participate in this research. If you decide at a later date that data should be destroyed we will honour your request in writing.

Pre-Questionnaire

PQ1. Position

PQ2. Institution

PQ3.
How relevant is your position related to Manuscriptology/Epigraphy?

- Not Relevant
- Mildly Relevant
- Somewhat Relevant
- Very Relevant
- Absolutely Relevant

PQ4.
How often do you handle manuscripts/epigraphs in your work (either directly or in terms of facsimiles/images)?

- Never
- Rarely
- Sometimes
- Very Often
- Always

PQ5.
How relevant to your work is identification/analysis of characters?

- Not Relevant
- Mildly Relevant

- Somewhat Relevant
- Very Relevant
- Absolutely Relevant

PQ6.
How often do you perform/require such analysis?

- Never
- Rarely
- Sometimes
- Very Often
- Always

PQ7.
Have you used any software for analyzing or studying manuscripts/epigraphs?

- Yes
- No

PQ8. If Yes, please explain briefly the software and the kind of analysis performed.

PQ9. How long do you use a computer for on an average week?

- < 10 hours
- 10 - 20 hours
- 20 - 30 hours
- > 30 hours

SAM. **Script Analysis Framework: Manual**

The application you are about to use is a prototype of a *script analysis framework*. It is intended to derive useful script-independent metrics from handwriting information of characters for descriptive and quantitative analysis. It consists of

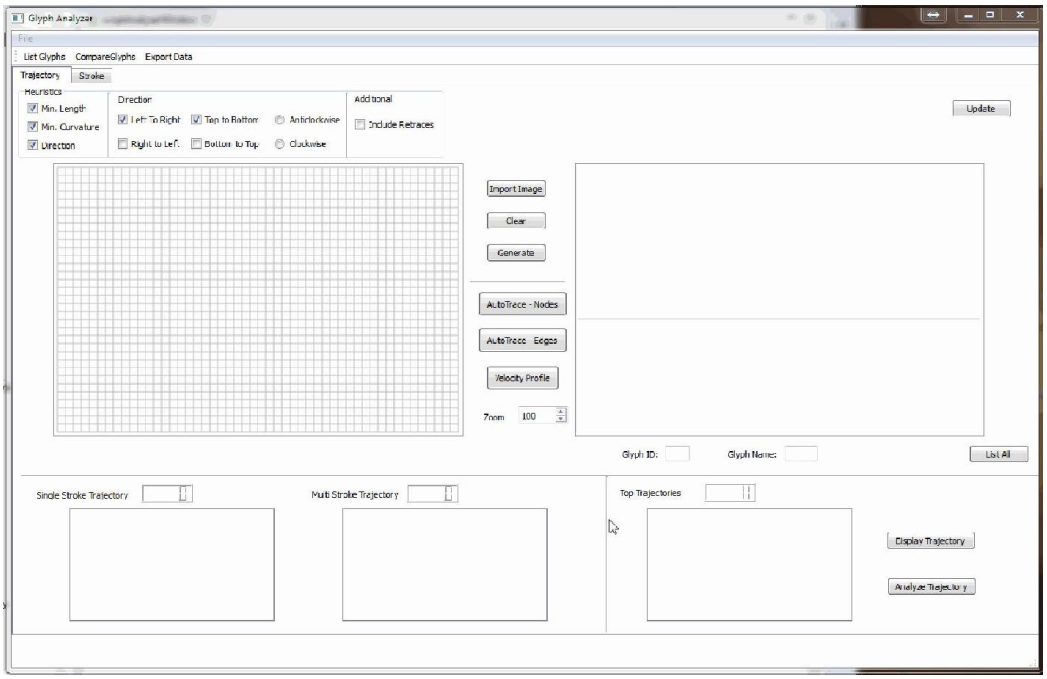
four modules:

Spline Conversion Trajectory Reconstruction Stroke Segmentation Feature Extraction

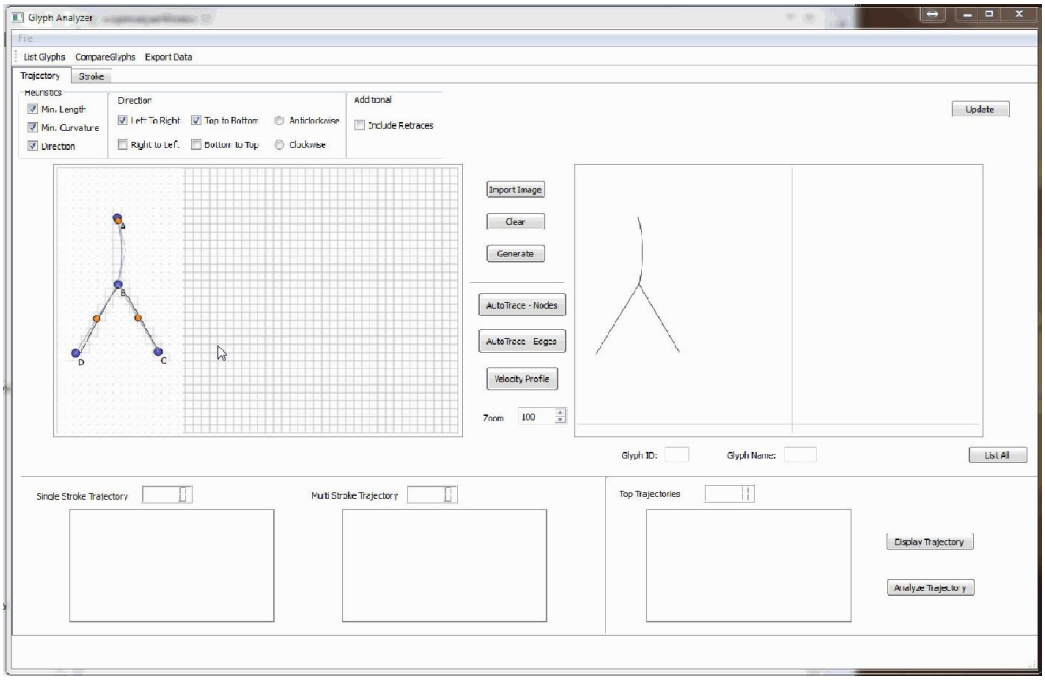
Spline Conversion:

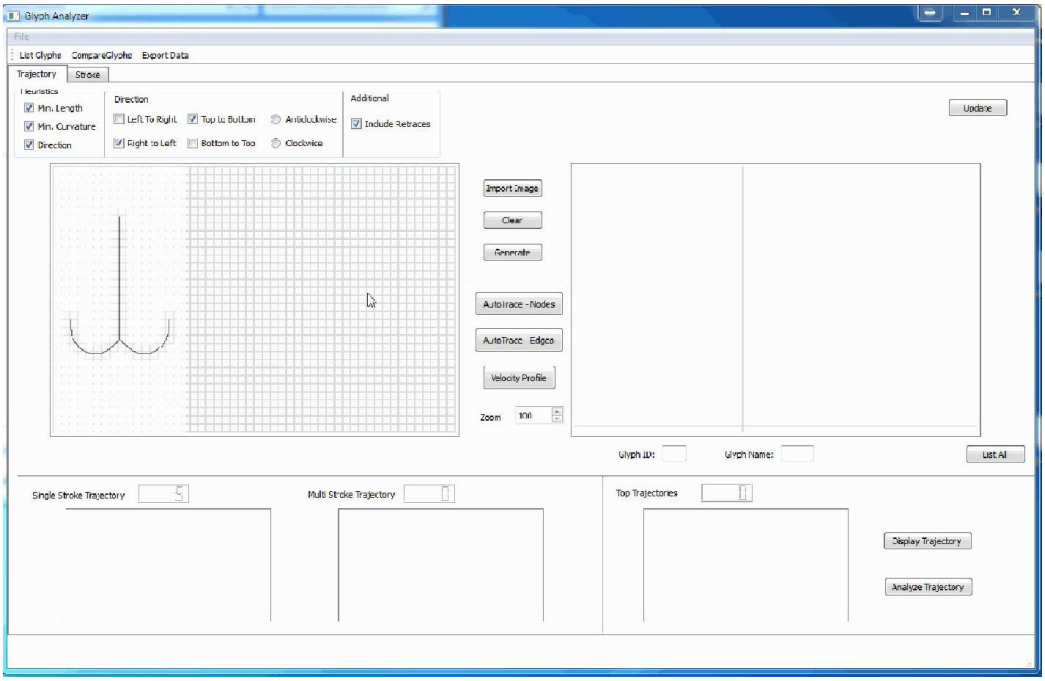
This module converts images to a computational representation for further analysis. This can be done automatically by importing images, or semi-automatically or even manually.

See below video for the automatic importing of images:

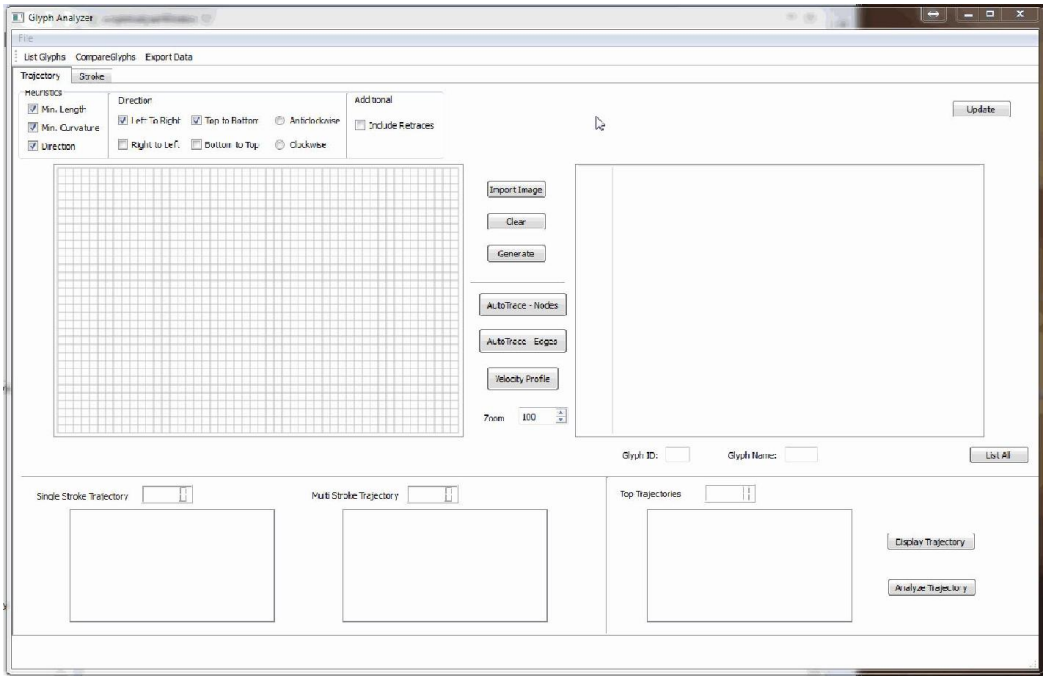


The results of the automatic process can also be changed if required. This is done by adjusting the orange circles. If you want to fine tune your adjustments, additional orange circles can be created by double-clicking the current orange circle.

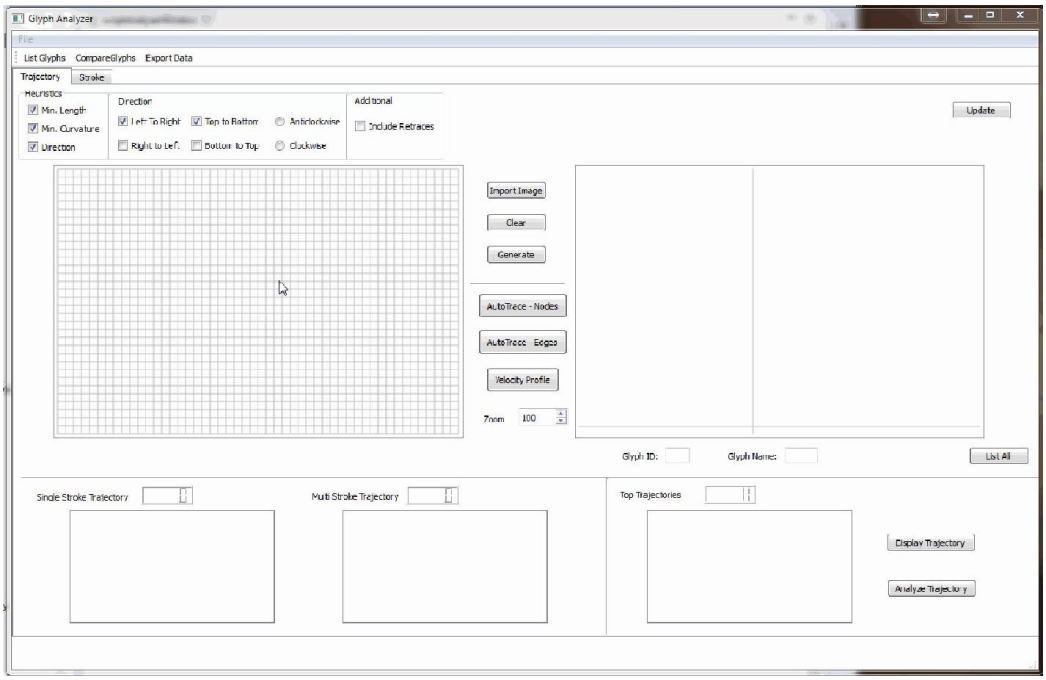




The computational representation can also be manually created with the imported image as a guidance. Double clicking on the left-hand panel creates nodes The edges between nodes can be created by successively double-clicking the required pair of nodes



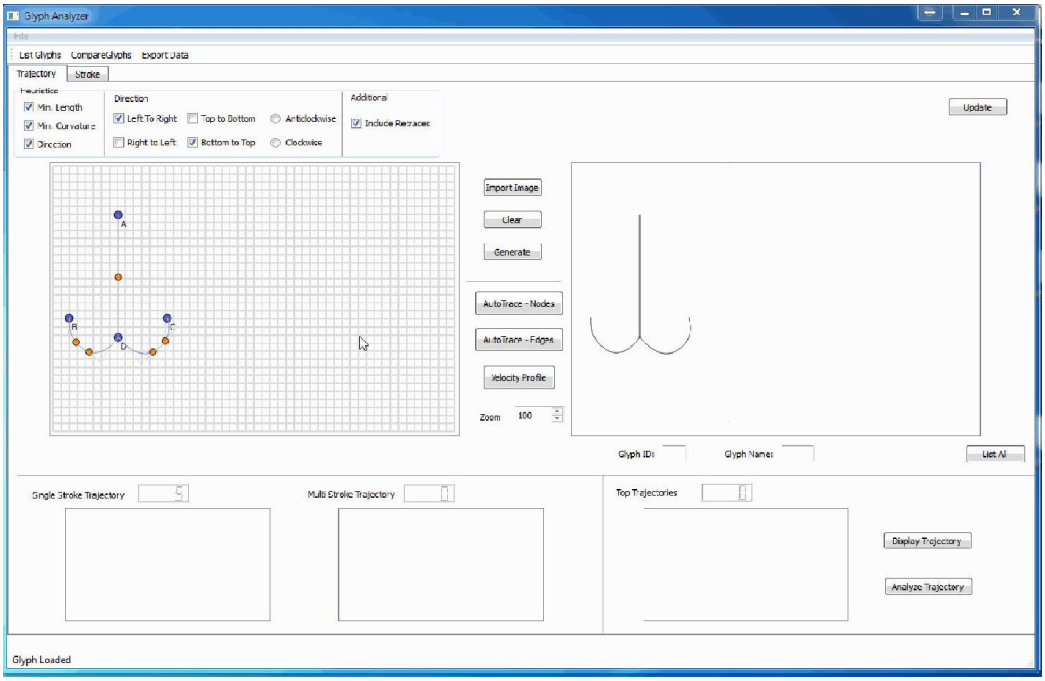
The shape may also be created by drawing on the right-hand panel. The glyph must be drawn while holding the left mouse button. This can be further refined manually as shown above.



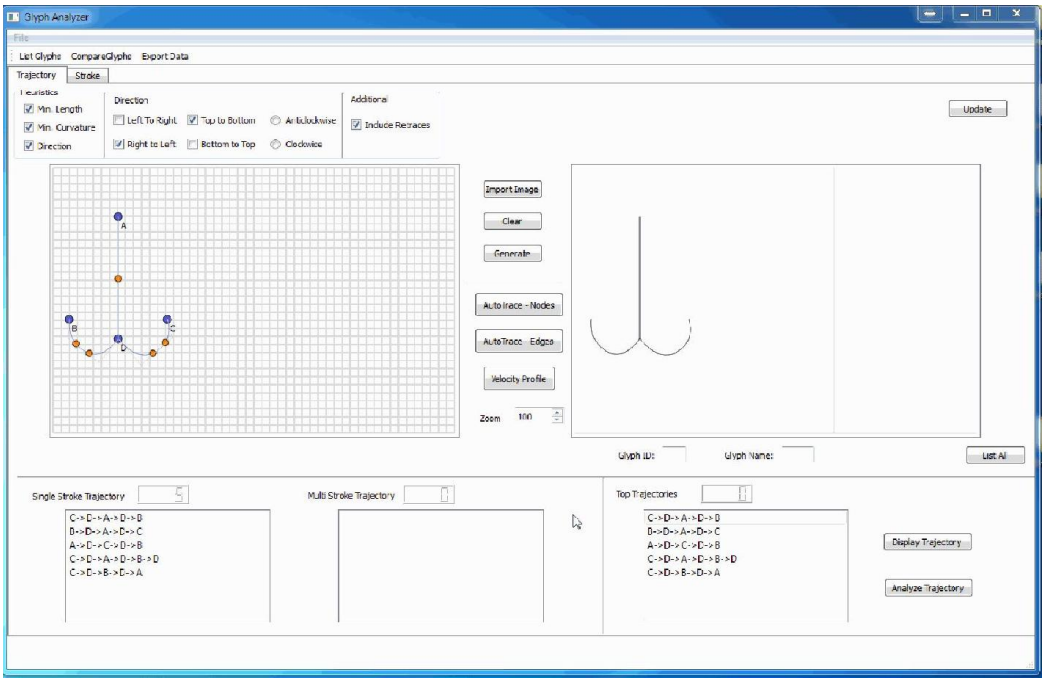
Trajectory Reconstruction

The trajectory indicates the actual hand movements required to write the character.

We reconstruct the trajectory using a heuristic process. The heuristics can also be adjusted using various options. The system will suggest five trajectories, ordering them based on their viability.



The user can also select one of those or manually override the trajectories as required. They need to double click the trajectory and change the path as required.



Stroke Segmentation
The characters are segmented into their handwriting “strokes". Strokes are short segments of a character that correspond to a basic hand movement.

Segmentation divides characters into two types of strokes.

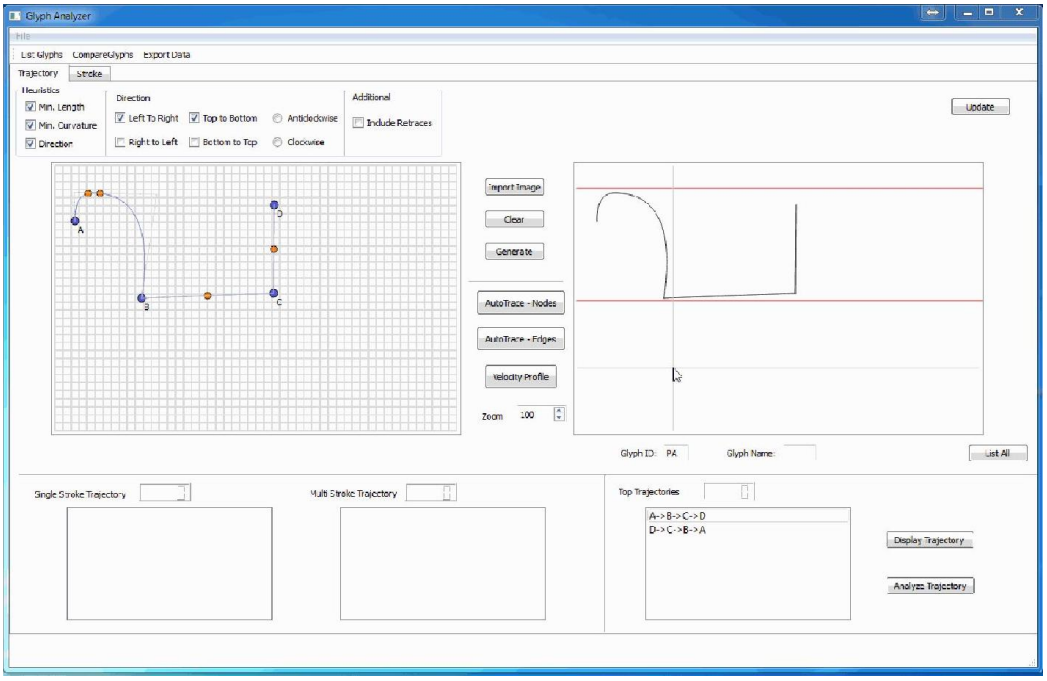
(i) Disjoint Strokes - strokes separated by a sharp junction (The left-hand panel)

The hand comes to near-complete stop, after writing these strokes

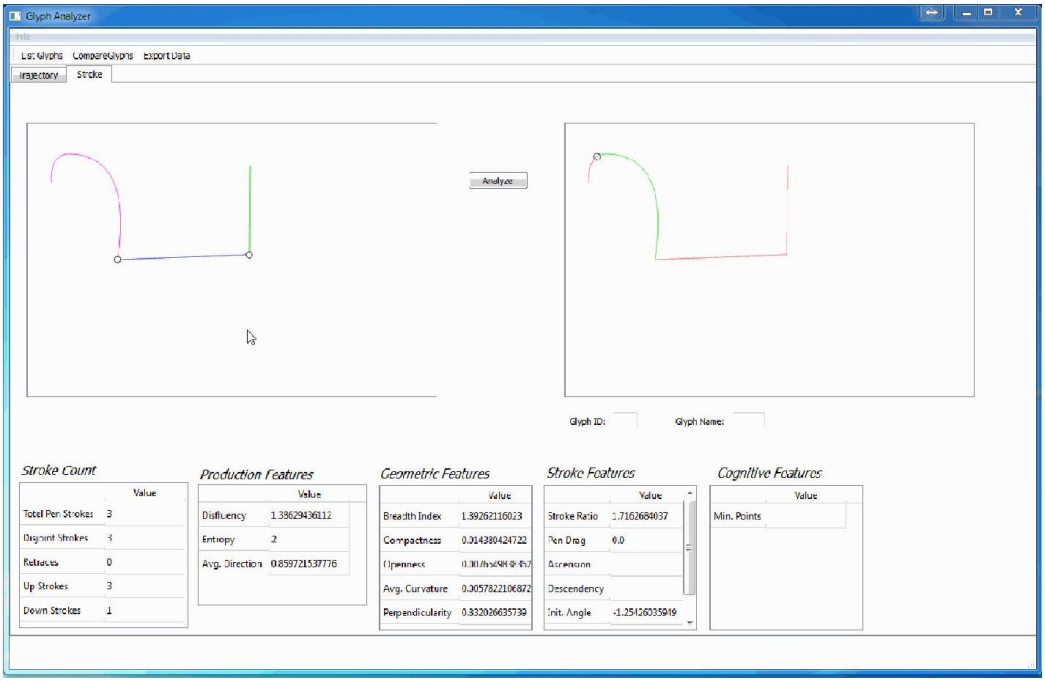
(ii) Primitive Strokes - strokes separated by a sharp curvature (The right hand panel)

The hand relatively slows down after writing these strokes.

This is performed by analyzing the trajectory of the character. Segmentation corresponding to a trajectory can be found by selecting the trajectory. Different trajectories yield correspondingly different segmentations. Select the trajectory which should be the basis of your segmentation.



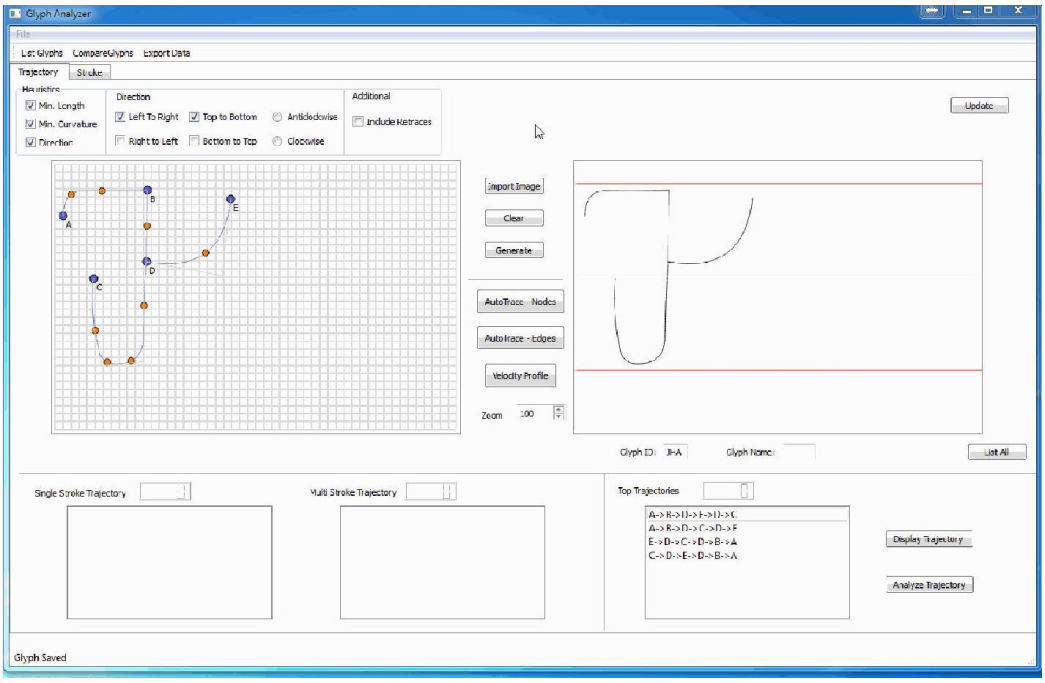
The segmentation can be overridden, by moving the nodes. New nodes can be inserted by double clicking on the right-hand panel.



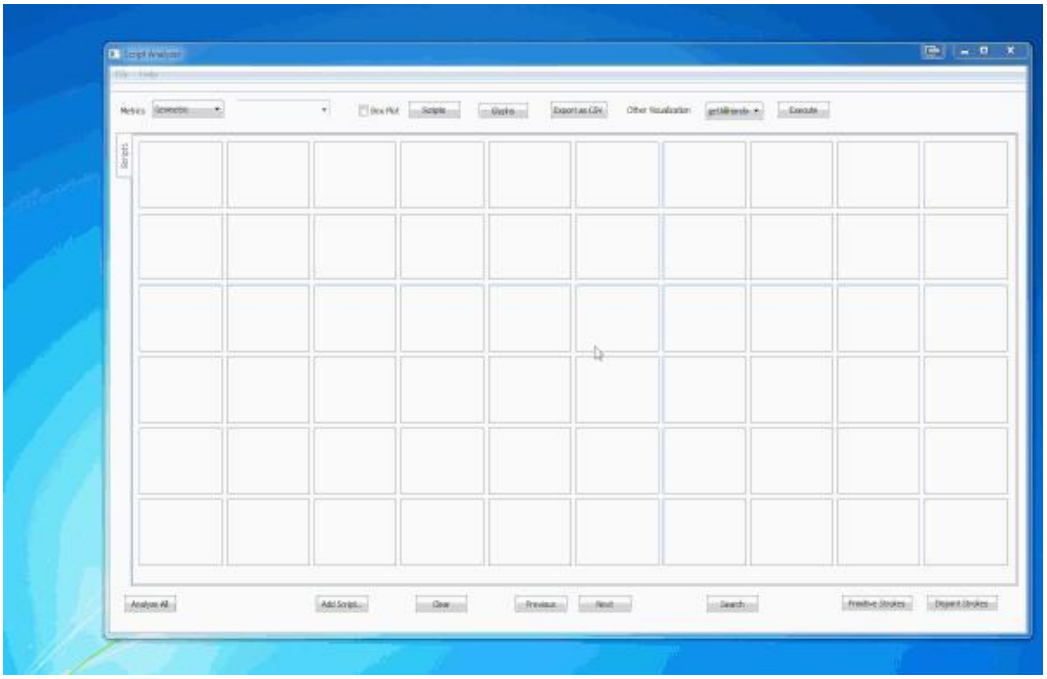
Feature Extraction

Based on the segmentation, features describing the appearance and writing of the characters are extracted. The text boxes in the above video display some of the metrics extracted from the glyph.

The entire analysis can be saved and also imported for later use.



The glyphs can be stored in a script repository. See below for script repository containing glyphs of several Indic scripts along with their analysis.



SAT.

Script Analysis Framework: Tasks

Please ask for assistance if required (You can also refer back to the documentation)

Task 1

For the first task, you'll be required to perform end-to-end analysis of a glyph from image import to feature extraction.

Please select up to 2 different characters and perform the complete analysis using the framework.

Task 2

For the second task:

1. Import an existing glyph and reconstruct the trajectory.
2. View various trajectories and try segmenting the character based on different trajectories.
3. You can also try changing the segmentation parameters

Please form this task for up to 5 different glyphs You can select and load from a set of existing glyphs. You can skip spline conversion part for this task. This has been done already.

Task 3

Using script repository, try loading different scripts and viewing already saved analysis.

Please view the trajectories and segmentations of at least 10 different characters.

SA1. General Usability

SA2. I think that I would like to use this system frequently.

Strongly Disagree	Disagree	Neither Agree nor Disagree	Agree	Strongly Agree	Not Applicable
○	○	○	○	○	○

SA3. I found the system unnecessarily complex.

Strongly Disagree	Disagree	Neither Agree nor Disagree	Agree	Strongly Agree	Not Applicable
○	○	○	○	○	○

SA4. I thought the system was easy to use.

Strongly Disagree	Disagree	Neither Agree nor Disagree	Agree	Strongly Agree	Not Applicable
○	○	○	○	○	○

SA5. I think that I would need the support of a technical person to be able to use this system.

Strongly Disagree	Disagree	Neither Agree nor Disagree	Agree	Strongly Agree	Not Applicable
○	○	○	○	○	○

SA6. I found the various functions in this system were well integrated.

Strongly Disagree	Disagree	Neither Agree nor Disagree	Agree	Strongly Agree	Not Applicable
○	○	○	○	○	○

SA7. I thought there was too much inconsistency in this system.

Strongly Disagree	Disagree	Neither Agree nor Disagree	Agree	Strongly Agree	Not Applicable
○	○	○	○	○	○

SA8. I would imagine that most people would learn to use this system very quickly.

Strongly Disagree	Disagree	Neither Agree nor Disagree	Agree	Strongly Agree	Not Applicable
○	○	○	○	○	○

SA9. I found the system very cumbersome to use.

Strongly Disagree	Disagree	Neither Agree nor Disagree	Agree	Strongly Agree	Not Applicable
○	○	○	○	○	○

SA10. I felt very confident using the system.

Strongly Disagree	Disagree	Neither Agree nor Disagree	Agree	Strongly Agree	Not Applicable
○	○	○	○	○	○

SA11. I needed to learn a lot of things before I could get going with this system.

Strongly Disagree	Disagree	Neither Agree nor Disagree	Agree	Strongly Agree	Not Applicable
○	○	○	○	○	○

SA12. **General Usefulness**
(Please answer the below questions in the context of analyzing/identifying characters)

SA13. Using the system would enable me to tasks more quickly

Strongly Disagree	Disagree	Neither Agree nor Disagree	Agree	Strongly Agree	Not Applicable
○	○	○	○	○	○

SA14. Using the system would improve my work performance

Strongly Disagree	Disagree	Neither Agree nor Disagree	Agree	Strongly Agree	Not Applicable

SA15. Using the system in my work would increase my productivity

Strongly Disagree	Disagree	Neither Agree nor Disagree	Agree	Strongly Agree	Not Applicable
○	○	○	○	○	○

SA16. Using the system would enhance my effectiveness on the work

Strongly Disagree	Disagree	Neither Agree nor Disagree	Agree	Strongly Agree	Not Applicable
○	○	○	○	○	○

SA17. Using the system would make it easier to do my work

Strongly Disagree	Disagree	Neither Agree nor Disagree	Agree	Strongly Agree	Not Applicable
○	○	○	○	○	○

SA18. I would find the system useful in my work

Strongly Disagree	Disagree	Neither Agree nor Disagree	Agree	Strongly Agree	Not Applicable
○	○	○	○	○	○

SA19. List positive aspects of the system

SA20. List negative aspects of the system

SA21.
Please arrange the modules in terms of importance

Spline Conversion
Trajectory Reconstruction
Stroke Segmentation
Feature Extraction

SA22.
I found the workflow of the system intuitive

Strongly Disagree	Disagree	Neither Agree nor Disagree	Agree	Strongly Agree	Not Applicable
○	○	○	○	○	○

SA23.
I found the ability to manually intervene/override very useful

Strongly Disagree	Disagree	Neither Agree nor Disagree	Agree	Strongly Agree	Not Applicable
○	○	○	○	○	○

SA24.
I felt the system allowed me to explore different kinds of analysis

Strongly Disagree	Disagree	Neither Agree nor Disagree	Agree	Strongly Agree	Not Applicable
○	○	○	○	○	○

SA25.

I felt the system allowed me a great deal of control with analysis

Strongly Disagree	Disagree	Neither Agree nor Disagree	Agree	Strongly Agree	Not Applicable
○	○	○	○	○	○

SA26.
Do you feel anything needs to be changed in terms of workflow?

- ○ Yes
- ○ No

SA27. If yes, Please explain

SA28.
Modules

SA29. Spline Conversion

	Strongly Disagree	Disagree	Neither Agree nor Disagree	Agree	Strongly Agree
I find this module very relevant to the workflow	○	○	○	○	○
I find the module very effective	○	○	○	○	○
I agree with the automatic analysis results of the module	○	○	○	○	○

SA30. Comments about the module (if any)

SA31. Trajectory Reconstruction

	Strongly Disagree	Disagree	Neither Agree nor Disagree	Agree	Strongly Agree
I find this module very relevant to the workflow					
I find the module very effective					
I agree with the automatic analysis results of the module					

SA32. Comments about the module (if any)

SA33. Stroke Segmentation

	Strongly Disagree	Disagree	Neither Agree nor Disagree	Agree	Strongly Agree
I find this module very relevant to the workflow					
I find the module very effective					
I agree with the automatic analysis results of the module					

SA34. Comments about the module (if any)

Q87. Please answer the below questions regarding data in the script repositories

	Strongly Agree	Agree	Neither Agree nor Disagree	Disagree	Strongly Disagree
The trajectory reconstructions of characters in the script repository are valid					
The segmentation of characters in the script repository are valid					

SA35. Any other comments/feedback about the overall system

MSM. Metrics of Characters

Visual Metrics

Length
This is the total length of the character. *Length* quantifies the entire movement of the pen required to produce the character.

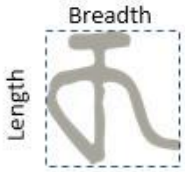
The length of the entire trajectory- 1-2-3-4-5 would be the length of the character

Divergence
Divergence is defined as distance between the starting position, where the writing instrument touches the writing surface and the ending position, where the instrument leaves the surface on completing the character. This metric quantifies the movement of the pen between those two events measuring how much the pen has visually 'diverged' from its original starting position. This is one of the important metrics that could be specific to a scribe.

Size
Size is measured by the bounding box area of a character. The bounding box is the minimal rectangle that encloses the given character. This could be directly correlated with the 'largeness' of the character and hence the term 'size'. The bigger the bounding box, the larger is the size of the character.

Length-Breadth Index

This is the ratio of the bounding box's height to the bounding box's width. This approximates the shape aspect of the character, i.e. slender/broad, etc.



Average Curvature
A straight stroke will have a curvature of zero compared to a curved stroke, which will have a higher curvature. Thus curved characters tend to have a higher average curvature compared to a characterwithless curves and/or more straight lines. This directly corresponds to the curvedness of the character.

Compactness
Compactness of a character is defined as the ratio between the length and the size. In some sense, it defines how compact (or dense) a character appears and directly corresponds to the number of strokes that a scribe is trying to fit within a given area. This makes it a very important metric to consider with characters as it very specific to a scribe or script. Some scribes may space out the character during production while others may tend to "compact" the strokes within a small area. This could be helpful in detection of hands in manuscripts.

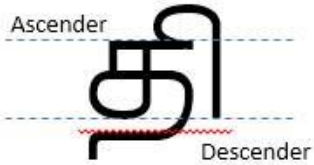
Figure on the left is more compact than the figure on the right.

Openness

Openness of a character can be defined as the ratio between divergence and length. This measures the movement of the pen with respect to its starting point and ending point and the length of the character. This may also be a scribe specific feature. A scribe may choose to 'close' a character irrespective of the length, while the some other scribe might just keep it open to save effort. In this case, the former will have low "openness" compared to the latter

Ascendance & Descendance

Some scripts have baselines. The percentages of the length of characters above and below baselines are defined as Ascendance andDescendance respectively.

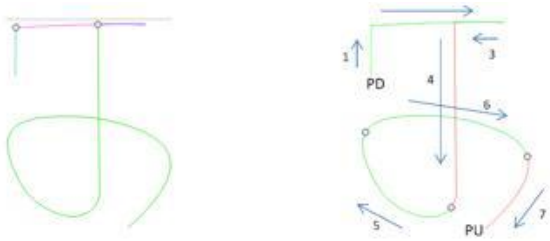


Circularity & Rectangularity

In many cases, the shapes of characters appear to approach an ideal geometric shape. We attempt to measure such approximations. Circularity and rectangularity could be defined as the deviation of the character's outline shape from that of an ideal circle and rectangle respectively.

Dynamic Information

Apart from the static shape of a character we also need also consider its dynamics. The character's kinematic (or temporal) information is essential in defining it. It dictates how the character is produced through the process of handwriting. Thus, deriving metrics quantifying properties of its production is very important.



Stroke Counts

A fundamental metric is to count the number of hand motions required to write characters. Humans generally attempt to minimize the number of hand-movements to write characters, but in some cases additional strokes are instead augmented into characters increasing the production effort. It is an interesting metric to analyze for the distribution across various scripts. This also allows us to study the human writing behavior by understanding the circumstances/environment under which such additions or reductions may occur.

Apart from the count of the primitive strokes, there are two more composite-stroke metrics that could be considered – pen-strokes & disjoint strokes. The former is the absolute hand movements required to write characters without a pen-up event and the latter are the composite-strokes that are delineated at sharp junctions.

The sample character has 1 pen stroke, 3 disjoint strokes, and 7 primitive strokes

Stroke Length

The distribution of the length of individual strokes and also calculating the average stroke length are very useful measures with respect to the analysis of writing. The average stroke length is a variable entity across different scripts or scribes and can be useful in classification.

Changeability

Handwriting consists of up-strokes and down-strokes. They are of two different characteristics with completely different physiological processes of production. It has been shown up-strokes are susceptible to change, while down-strokes are invariant and more stable. Up-strokes are faster (and hence perhaps less stable. Strokes that are produced between 210° and 280° to be down-strokes and all non-down strokes are included as up-strokes. So the ability of the character to change, i.e. *changeability,* can be directly tied to the ratio of up-strokes' length to that of the down-strokes' length. If the ratio is high the character can be considered susceptible to change. Thus changeability as a metric is related to a character's susceptibility to change.

Disfluency

It is known that handwriting fluency is affected at points where curvature is at its maximum/minimum. The transition between down-strokes and up-strokes is also considered to slow down the writing process. The number of sharp junctions in a character also contributes to the slowing of velocity during the handwriting production. The sum of all points that affect velocity is termed as *disfluency*. Based on our previously stated assumptions elsewhere these would be *points of high curvature,, sharp-junctions,* and *intermediate pen-up events*. This can directly correspond to the difficulty in terms of writing the character. A character with higher number of disfluent points is harder to produce as the velocity is frequently interrupted.

For instance, the sample character will be thought of have a disfluency of 6.

Entropy

In information theory, entropy is defined as the average amount of information contained within an entity. This amount of information in the system is directly proportional to the randomness or disorderliness present in the system. When there are several instances of change, it results in an increase of entropy as it contains more information. To calculate the entropy of a character the trajectory of the character is 'quantized' into chain codes denoting the major eight directions. Assigning a chain code to the individual strokes performs this. The eight chain codes correspond to the following directions - N, S, E, W, NE, NW, SE, and SW.

The sample character in the above figure can be quantized into [N E W S NW SE SW].

Any character with a sufficient number of repeating patterns will record low entropy and those with no patterns high entropy. Thus the entropy of characters conveys the randomness associated with the pen movements required to produce the character.

N-Gram model of scripts

Writing a character can be very well considered to be similar to that of constructing a sentence. While sentences are made up of words, characters are made of strokes. We here seek to apply some aspects of natural language processing to scripts. N-gram modeling is frequently used in natural language processing for a wide variety of purposes. N-gram model is a probabilistic model to predict the next item in a sequence. It is now possible to calculate the entropy of a script as opposed to that of a character and also allows us to study the regularity of stroke combinations.

Angle-Based Metrics

Analyzing the different angles of strokes occurring in scripts can throw more light on a particular scribal behavior. We define a few important angle-based metrics that could be used. *Major angle* is the angle of the largest primitive stroke present in the character. The *initial angle* is defined with the initial stroke of a character. The *divergence angle* - angle between first and last points could also be considered as a metric. For multi-stroke characters, *angle of pen-drag* can be an important measure. *Inter-stroke angles* can be plotted as a histogram to see the changes.

Pen-Drag Distance

The *pen-drag distance* is a metric with respect to multi-stroke behavior. This captures the hand movements between pen-strokes, which are an important part of multi-stroke production.

Cognitive Metrics

Writing a character is usually a top-down process. A character has to be memorized and then reproduced. Consequently, this requires elaborate trajectory planning. We need to find out the approximate information required to cognitively memorize and produce characters.

We attempt to find out the minimal representation of a character required to reproduce it. Theoretically, these would be the points necessary to plan the trajectory of the character. In fact, this directly corresponds to the 'landmark points' [Points of high curvature, disjoint points] in the character, as those are the points that define its shape.

This metric mostly corresponds to the cognitive information present in a character and could be used as required.

MSM. Please answer the below questions in the context of analyzing/identifying characters

MSM1. I find the metric very relevant

	Strongly Disagree	Disagree	Neither Agree nor Disagree	Agree	Strongly Agree
Visual: Length					
Visual: Divergence					
Visual: Size					
Visual: Length-Breadth Index					
Visual: Average Curvature					
Visual: Compactness					
Visual: Openness					
Visual: Ascendance & Descandance					
Visual: Circularity & Rectangularity					
Dynamic: Stroke Counts					
Dynamic: Stroke Length					
Dynamic: Changeability					
Dynamic: Disfluency					
Dynamic: Entropy					
Dynamic: N-Gram model of scripts					
Dynamic: Angle-based metrics					
Dynamic: Pen-Drag distance					
Cognitive: Number of landmark points					

MSM2.
I find the metric very useful

	Strongly Disagree	Disagree	Neither Agree nor Disagree	Agree	Strongly Agree
Visual: Length					
Visual: Divergence					
Visual: Size					
Visual: Length-Breadth Index					
Visual: Average Curvature					
Visual: Compactness					
Visual: Openness					
Visual: Ascendance & Descandance					
Visual: Circularity & Rectangularity					
Dynamic: Stroke Counts					
Dynamic: Stroke Length					
Dynamic: Changeability					
Dynamic: Disfluency					
Dynamic: Entropy					
Dynamic: N-Gram model of scripts					
Dynamic: Angle-based metrics					
Dynamic: Pen-Drag distance					
Cognitive: Number of landmark points					

MSM3.
I would use the metric in the future

	Strongly Disagree	Disagree	Neither Agree nor Disagree	Agree	Strongly Agree
Visual: Length					
Visual: Divergence					
Visual: Size					
Visual: Length-Breadth Index					
Visual: Average					

Curvature					
Visual: Compactness					
Visual: Openness					
Visual: Ascendance & Descandance					
Visual: Circularity & Rectangularity					
Dynamic: Stroke Counts					
Dynamic: Stroke Length					
Dynamic: Changeability					
Dynamic: Disfluency					
Dynamic: Entropy					
Dynamic: N-Gram model of scripts					
Dynamic: Angle-based metrics					
Dynamic: Pen-Drag distance					
Cognitive: Number of landmark points					

MSM4.
I would use the metric frequently

	Strongly Disagree	Disagree	Neither Agree nor Disagree	Agree	Strongly Agree
Visual: Length					
Visual: Divergence					
Visual: Size					
Visual: Length-Breadth Index					
Visual: Average Curvature					
Visual: Compactness					
Visual: Openness					
Visual: Ascendance & Descandance					
Visual: Circularity & Rectangularity					
Dynamic: Stroke Counts					

Dynamic: Stroke Length					
Dynamic: Changeability					
Dynamic: Disfluency					
Dynamic: Entropy					
Dynamic: N-Gram model of scripts					
Dynamic: Angle-based metrics					
Dynamic: Pen-Drag distance					
Cognitive: Number of landmark points					

MSM5. What kind of analysis would your perform using these metrics?

MSM6. Order the metrics in terms of importance

Visual: Length
Visual: Divergence
Visual: Size
Visual: Length-Breadth Index
Visual: Average Curvature
Visual: Compactness
Visual: Openness
Visual: Ascendance & Descandance
Visual: Circularity & Rectangularity
Dynamic: Stroke Counts
Dynamic: Stroke Length

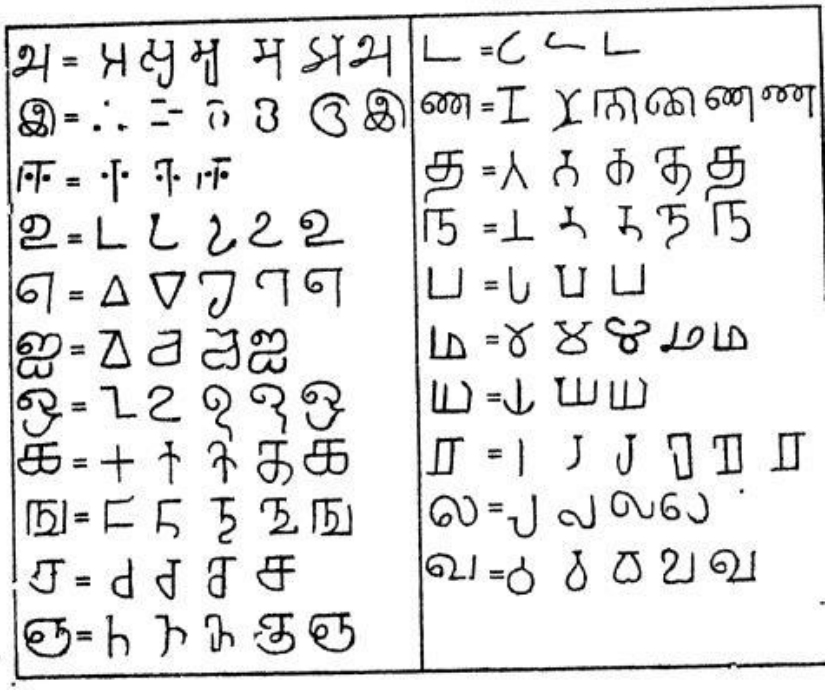
Dynamic: Changeability
Dynamic: Disfluency
Dynamic: Entropy
Dynamic: N-Gram model of scripts
Dynamic: Angle-based metrics
Dynamic: Pen-Drag distance
Cognitive: Number of landmark points

MSM7. Any other comments/feedback

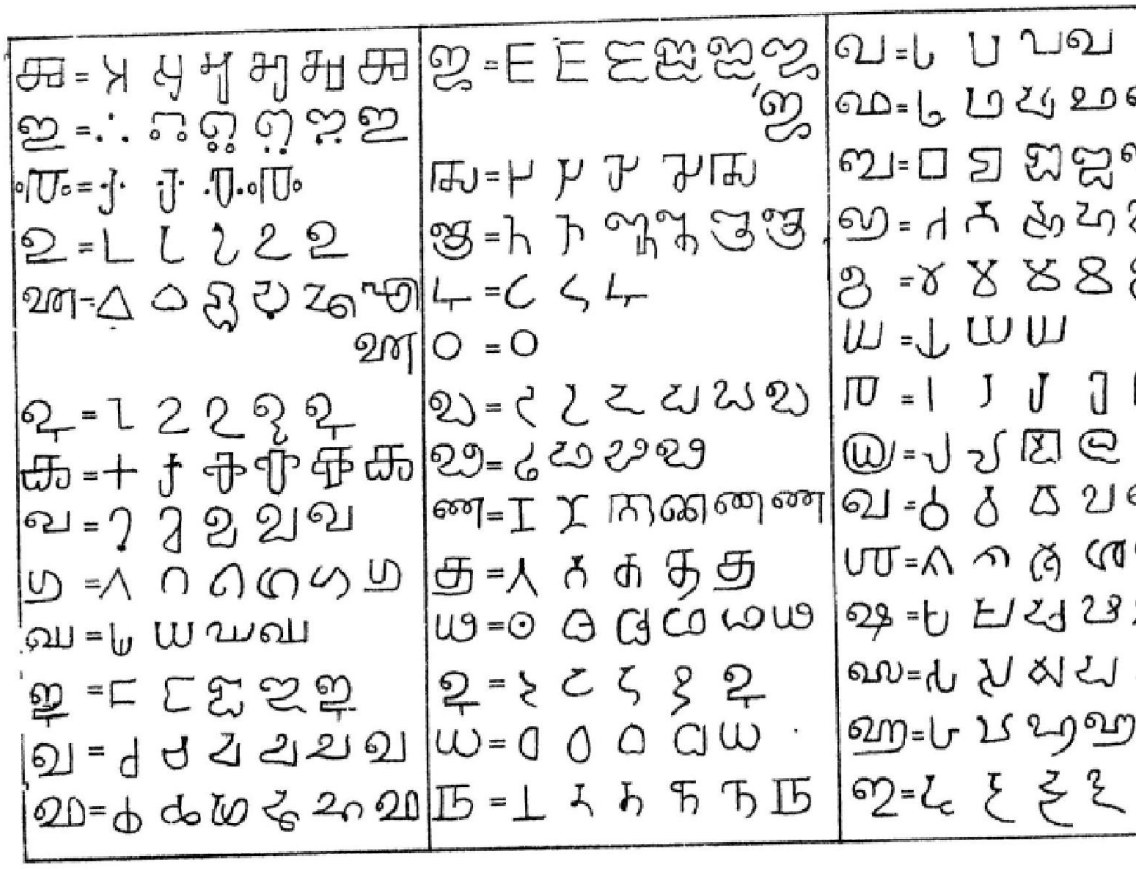
DIS. **Development of Indic Scripts**

We present below the evolution of 4 Major Indic scripts (Tamil, Grantha, Kannada & Devanagari) from Brahmi script.

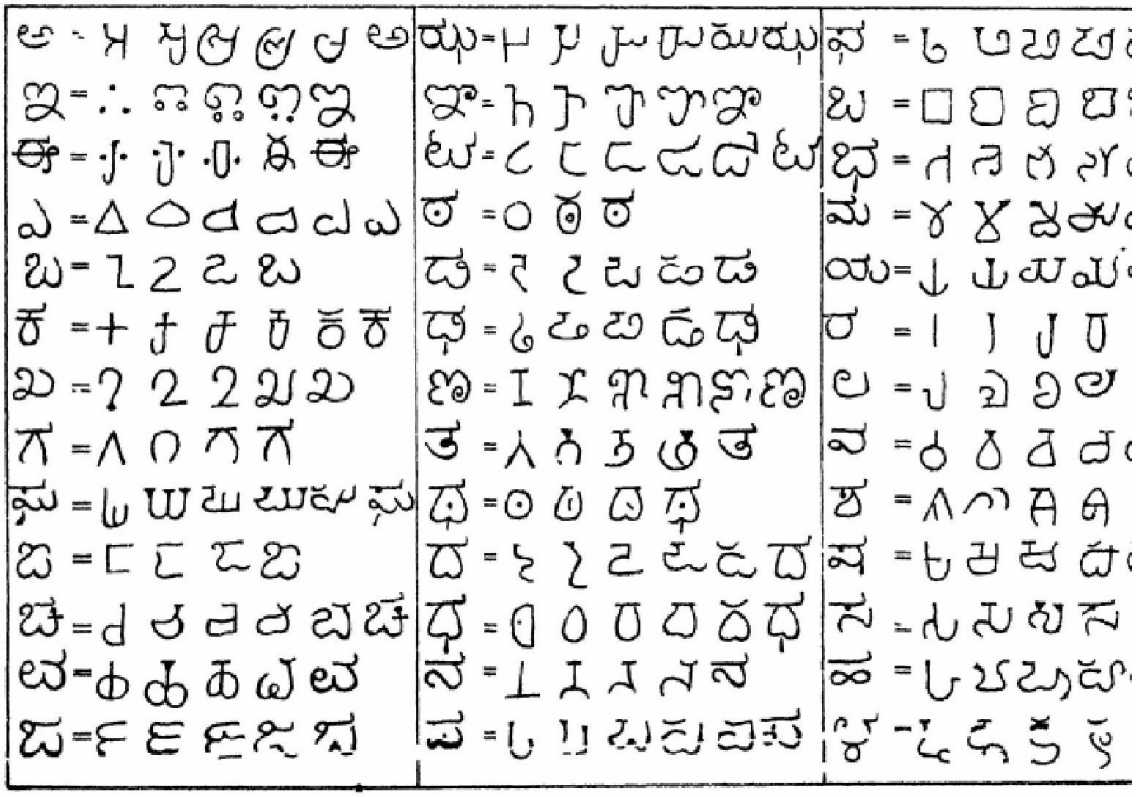
Tamil



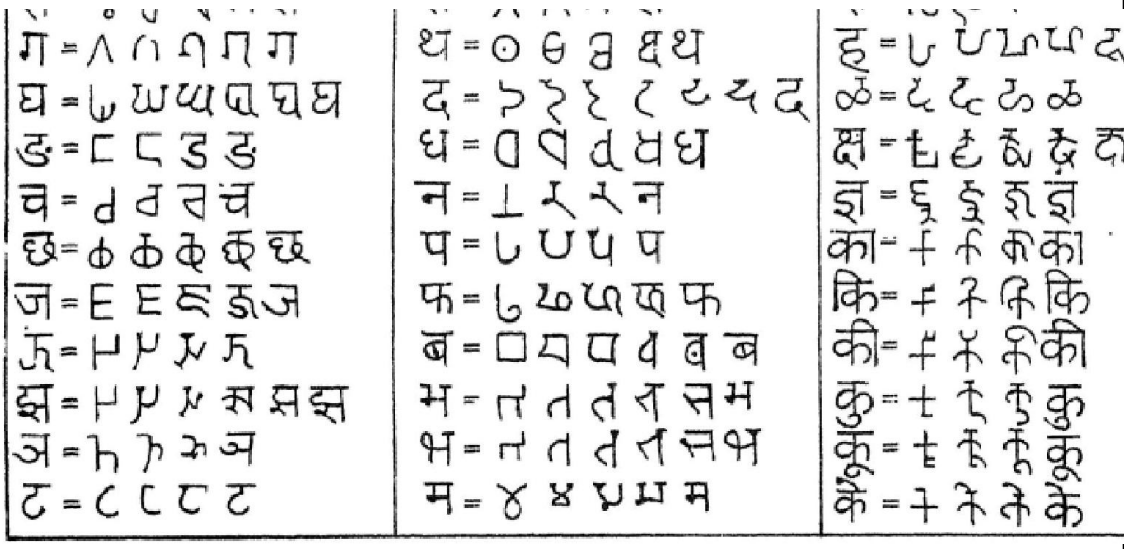
Grantha



Kannada



Devanagari



We have used the earlier discussed metrics to derive some results to understand the development of the modern varieties of these scripts. (Please refer to the metrics in the previous sheet if required)

DIS1.
Please let us know if you agree with the following statements

DIS2.
The below statements point to the visual attributes of the characters

DIS3.
Tamil

	Strongly Disagree	Disagree	Neither Agree nor Disagree	Agree	Strongly Agree
The length of					

characters have increased over time					
The length-breadth index of the characters have increased over time					
The circularity of the characters have increased over time					
The rectangularity of the characters have increased over time					
The openness of the characters have decreased over time					
The compactness of the characters has decreased over time					
The average curvature of the characters have increased over time					
The characters have become more complex to write over time					

DIS4.
Grantha

	Strongly Disagree	Disagree	Neither Agree nor Disagree	Agree	Strongly Agree
The length of characters have increased over time					
The length-breadth index of the characters have increased over time					
The circularity of the characters have increased over time					
The rectangularity of the characters have increased over time					
The openness of the characters have decreased over time					
The compactness of the characters has decreased over time					

The average curvature of the characters have increased over time					
The characters have become more complex to write over time					

DIS5.
Kannada

	Strongly Disagree	Disagree	Neither Agree nor Disagree	Agree	Strongly Agree
The length of characters have increased over time					
The length-breadth index of the characters have increased over time					
The circularity of the characters have increased over time					
The rectangularity of the characters have increased over time					
The openness of the characters have decreased over time					
The compactness of the characters has decreased over time					
The average curvature of the characters have increased over time					
The characters have become more complex to write over time					

DIS6.
Devanagari

	Strongly Disagree	Disagree	Neither Agree nor Disagree	Agree	Strongly Agree
The length of					

characters have increased over time					
The length-breadth index of the characters have increased over time					
The circularity of the characters have increased over time					
The rectangularity of the characters have increased over time					
The openness of the characters have decreased over time					
The compactness of the characters has decreased over time					
The average curvature of the characters have increased over time					
The characters have become more complex to write over time					

DIS7. The below statements point to the various factors that mainly changed (thus causing diversification) in the development of the four scripts that had been shown earlier.

	Strongly Disagree	Disagree	Neither Agree nor Disagree	Agree	Strongly Agree
The characters diversified based on Compactness					
The characters diversified based on Average Curvature					
The characters diversified based on Circularity					
The characters diversified based on Rectangularity					

DIS8. The below statements point to how variations have spread to shape the characters of the scripts in their current form.

	Strongly Disagree	Disagree	Neither Agree nor Disagree	Agree	Strongly Agree
Brahmi characters are very similar to each other compared to other characters					
The characters gained more symmetry, length and less openness [in the intermediate stage]					
The characters at the end stage gained more symmetry and curvature [in their final form]					
The characters at the end stage gained more length and less openness [in their final form]					
I see a specific pattern in the way characters have evolved					

DIS9.
The characters within each script have become similar to each other over time

Strongly Disagree	Disagree	Neither Agree nor Disagree	Agree	Strongly Agree

DIS10.
I find quantitative analysis such as this interesting

Strongly Disagree	Disagree	Neither Agree nor Disagree	Agree	Strongly Agree

DIS11.
I will perform similar quantitative analysis in the future

Strongly Disagree	Disagree	Neither Agree nor Disagree	Agree	Strongly Agree

DIS12. Any other comments/feedback

References

Aiolli, F. & Ciula, A. (2009). A case study on the System for Paleographic Inspections (SPI): challenges and new developments. In *Computational Intelligence and Bioengineering: Essays in memory of Antonina Starita, Proceedings of the 2009 conference on* (pp. 53–66).

Aksentijevic, A. & Gibson, K. (2012). Psychological complexity and the cost of information processing. *Theory & Psychology*, *22*(5), 572–590.

Almaksour, A., Anquetil, E., Plamondon, R. & O'Reilly, C. (2011). Synthetic handwritten gesture generation using Sigma-Lognormal model for evolving handwriting classifiers. In *International Graphonomics Society, 15th Biennial Conference of the.*

Altmann, G. (2004). Script complexity. *Glottometrics 8, 68, 74*.

Altmann, G. & Fengxiang, F. (2008). *Analyses of script: properties of characters and writing systems* (Vol. 63). Walter de Gruyter.

Antić, G. & Altmann, G. (2005). On letter distinctivity. *Glottometrics 9, 46, 53*.

Aussems, M. (2006). Christine de Pizan and the scribal fingerprint: a quantitative approach to manuscript studies. *Utrecht: unpublished Research Master's Thesis*.

Aussems, M. & Brink, A. (2009). Digital palaeography. *Kodikologie und Paläographie im Digitalen Zeitalter–Codicology and Palaeography in the Digital Age*, 293–308.

Azmi, M. S., Omar, K., Nasrudin, M. F., Ghazali, K. W. M. & Abdullah, A. (2011). Arabic calligraphy identification for Digital Jawi Paleography

using triangle blocks. In *Electrical Engineering and Informatics (ICEEI), 2011 International Conference on* (pp. 1–5).

Bangor, A., Kortum, P. T. & Miller, J. T. (2008). An empirical evaluation of the system usability scale. *Intl. Journal of Human–Computer Interaction*, *24*(6), 574–594.

Berio, D. & Leymarie, F. F. (2015). Computational models for the analysis and synthesis of graffiti tag strokes. In *Workshop on Computational Aesthetics, Proceedings of the* (pp. 35–47).

Bezine, H., Alimi, A. M. & Sherkat, N. (2004). Generation and analysis of handwriting script with the beta-elliptic model. In *Frontiers in Handwriting Recognition, 9th International Workshop on* (pp. 515–520).

Bhat, A. & Hammond, T. (2009). Using Entropy to Distinguish Shape versus Text in Hand-Drawn Diagrams. In *IJCAI* (Vol. 9, pp. 1395–1400).

Bischoff, B. & Koch, W. (1979). *Paläographie des römischen Altertums und des abendländischen Mittelalters*. E. Schmidt.

Bishop, T. A. M. (1961). *Scriptores regis*. Clarendon Press.

Blanchard, A. (1999). L'hypothèse de l'unité de ductus en paléographie papyrologique. *Scrittura é Civiltà*(23), 5–27.

Brink, A., Smit, J., Bulacu, M. & Schomaker, L. (2012). Writer identification using directional ink-trace width measurements. *Pattern Recognition*, *45*(1), 162–171.

Brooke, J. et al. (1996). SUS-a quick and dirty usability scale. *Usability evaluation in industry*, *189*(194), 4–7.

Brown, M. P. (1996). *The Book of Cerne: prayer, patronage, and power in ninth-century England* (Vol. 1). University of Toronto Press.

Brown, T. J. (1993). *A Palaeographer's View: Selected Writings.* Ed. JM Bately, MP Brown and J. Roberts. London: Harvey Miller Publishers.

Bulacu, M. & Schomaker, L. (2006). Combining multiple features for text-independent writer identification and verification. In *Frontiers in Handwriting Recognition, 10th International Workshop on.*

Bulacu, M. & Schomaker, L. (2007a). Automatic handwriting identification on medieval documents. In *Image Analysis and Processing (ICIAP 2007), Proceedings of the 14th International Conference on* (pp. 279–284).

Bulacu, M. & Schomaker, L. (2007b). Text-independent writer identification and verification using textural and allographic features. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, *29*(4), 701–717.

Bulacu, M., Schomaker, L. & Vuurpijl, L. (2003). Writer identification using edge-based directional features. In *Document Analysis and Recognition, Proceedings of the 7th International Conference on* (p. 937).

Bunke, H., Ammann, R., Kaufmann, G., Ha, T. M., Schenkel, M., Seiler, R. & Eggimann, F. (1997). Recovery of temporal information of cursively handwritten words for on-line recognition. In *Document Analysis and Recognition, Proceedings of the 4th International Conference on* (Vol. 2, pp. 931–935).

Burgers, J. W. J. et al. (1995). *De paleografie van de documantaire bronnen in Holland en Zeeland in de dertiende eeuw*. Peeters.

Canart, P. (2006). La paléographie est-elle un art ou une science? *Scriptorium*, *60*(2), 159–185.

Changizi, M. A. & Shimojo, S. (2005). Character complexity and redundancy in writing systems over human history. *Royal Society of London B: Biological Sciences, Proceedings of the*, *272*(1560), 267–275.

Changizi, M. A., Zhang, Q., Ye, H. & Shimojo, S. (2006). The structures of letters and symbols throughout human history are selected to match those found in objects in natural scenes. *The American Naturalist*, *167*(5), E117–E139.

Chen, J., Zou, L.-h., Zhang, J. & Dou, L.-h. (2009). The comparison and application of corner detection algorithms. *Journal of Multimedia*, *4*(6), 435–441.

Choi, H., Cho, S.-J. & Kim, J. H. (2003). Generation of handwritten characters with Bayesian network based on-line handwriting recognizers. In *null* (p. 995).

Ciula, A. (2005). Digital palaeography: using the digital representation of medieval script to support palaeographic analysis. *Digital Medievalist*, *1*(1).

Ciula, A. (2009). The palaeographical method under the light of a digital approach. *Kodikologie und Paläographie im digitalen Zeitalter-Codicology and Palaeography in the Digital Age*, 219–237.

Costamagna, G., Gilisse, L., Gasparri, F. & Pratesi, A. (1995). Commentare Bischoff. *Scrittura e civiltà*, *19*, 321–352.

Daston, L. & Galison, P. (2007). Objectivity. brooklyn. *NY: Zone Books*.

Davis, F. D. (1989). Perceived usefulness, perceived ease of use, and user acceptance of information technology. *MIS quarterly*, 319–340.

Davis, T. (2007). The practice of handwriting identification. *The Library*, *8*(3), 251–276.

De Boor, C. (1978). *A practical guide to splines* (Vol. 27). Springer-Verlag New York.

Delaye, A. & Anquetil, E. (2013). HBF49 feature set: A first unified baseline for online symbol recognition. *Pattern Recognition*, *46*(1), 117–130.

Denier, J. & Thuring, J. P. (1965). The guiding of human writing movements. *Biological Cybernetics*, *2*(4), 145–148.

Derolez, A. (2003). *The palaeography of Gothic manuscript books: from the twelfth to the early sixteenth century* (Vol. 9). Cambridge University Press.

d'Haenens, A. (1975). I. pour une sémiologie paléographique et une histoire de l'écriture. *Scriptorium*, *29*(2), 175–198.

Djioua, M., O'Reilly, C. & Plamondon, R. (2006). An interactive trajectory synthesizer to study outlier patterns in handwriting recognition and signature verification. In *Pattern Recognition, 18th International Conference on* (Vol. 1, pp. 1124–1127).

Djioua, M. & Plamondon, R. (2008). An interactive system for the automatic generation of huge handwriting databases from a few specimens. In *Pattern Recognition, 19th International Conference on* (pp. 1–4).

Djioua, M. & Plamondon, R. (2009). A new algorithm and system for the characterization of handwriting strokes with delta-lognormal parameters. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, *31*(11), 2060–2072.

Doermann, D. S. & Rosenfeld, A. (1995). Recovery of temporal information from static images of handwriting. *International Journal of Computer Vision*, *15*(1-2), 143–164.

Dolinskỳ, J. & Takagi, H. (2007). Synthesizing handwritten characters

using naturalness learning. In *Computational Cybernetics. IEEE International Conference on* (pp. 101–106).

Douglas, D. H. & Peucker, T. K. (1973). Algorithms for the reduction of the number of points required to represent a digitized line or its caricature. *Cartographica: The International Journal for Geographic Information and Geovisualization*, *10*(2), 112–122.

Edelman, S. & Flash, T. (1987). A model of handwriting. *Biological cybernetics*, *57*(1-2), 25–36.

Eden, M. (1968). Handwriting generation and recognition. *Recognizing patterns*, 138–154.

Fink, G. A. (2014). Markov models for pattern recognition: from theory to applications. In (pp. 107–127). Springer Science & Business Media.

Fowler, H. N. (1925). Plato in Twelve Volumes, Vol. 9. *Cambridge, MA, Harvard University Press*.

Freeman, H. (1974). Computer processing of line-drawing images. *ACM Computing Surveys (CSUR)*, *6*(1), 57–97.

Gao, D.-G. & Kao, H. S. (2002). Psycho-geometric analysis of commonly used Chinese characters. *Cognitive neuroscience studies of the Chinese language*, 195–206.

Gilissen, L. (1973). L'expertise des écritures médiévales. *Recherche d'une méthode avec application à un manuscrit du XIe siècle: Le Lectionnaire de Lobbes, Codex Bruxellensis*, *18018*, 20–32.

Gumbert, J. P. (1976). A proposal for a Cartesian nomenclature. *Essays presented to GI Lieftinck, IV: miniatures, scripts, collections*, 45–52.

Hahn, U., Chater, N. & Richardson, L. B. (2003). Similarity as transformation. *Cognition*, *87*(1), 1–32.

Hanssen, A., Leeuwenberg, E. & vd Helm, P. (1993). Metrical information load of lines and angles in line patterns. *Psychological research*, *55*(3), 191–199.

Hassner, T., Rehbein, M., Stokes, P. A. & Wolf, L. (2013). Computation and Palaeography: Potentials and Limits (Dagstuhl Perspectives Workshop 12382). *Dagstuhl Manifestos*, *2*(1).

Hassner, T., Sablatnig, R., Stutzmann, D. & Tarte, S. (2014). Digital Palaeography: New Machines and Old Texts (Dagstuhl Seminar 14302). *Dagstuhl Reports*, *4*(7).

Hegenbarth-Reichardt, I. & Altmann, G. (2008). On the decrease of complexity from hieroglyphs to hieratic symbols. *Analyses of Script: Properties of Characters and Writing Systems*, 101–110.

Herzog, R., Neumann, B. & Solth, A. (2010). Computer-based Stroke Extraction in Historical Manuscripts. *Technical Report: Report FBI-HH-B-296/10, University of Hamburg, Computer Science Departement, The Cognitive Systems Laboratory*.

Hollerbach, J. M. (1981). An oscillation theory of handwriting. *Biological Cybernetics*, *39*(2), 139–156.

Impedovo, D. & Pirlo, G. (2008). Automatic signature verification: the state of the art. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, *38*(5), 609–635.

Isokoski, P. (2001). Model for unistroke writing time. In *SIGCHI conference on Human Factors in Computing Systems, Proceedings of the* (pp. 357–364).

Jäger, S. (1996). Recovering writing traces in off-line handwriting recognition: Using a global optimization technique. In *Pattern Recognition, Proceedings of the 13th International Conference on* (Vol. 3, pp. 150–154).

Köhler, R. (2008). Quantitative analysis of writing systems: an introduction. *Analyses of script: properties of characters and writing systems*, 3–9.

Koster, W. & Vredenbregt, J. (1971). *Analysis and synthesis of handwriting*. Karger Publishers.

Kumar, D. U., Sreekumar, G. & Athvankar, U. (2009). Traditional writing system in southern indiaâĂŤpalm leaf manuscripts. *Design Thoughts*, *7*.

Lallican, P. M. & Viard-Gaudin, C. (1997). A Kalman approach for stroke order recovering from off-line handwriting. In *Document Analysis and Recognition, Proceedings of the 4th International Conference on* (Vol. 2, pp. 519–522).

Lau, K. K., Yuen, P. C. & Tang, Y. Y. (2003). Recovery of writing sequence of static images of handwriting using UWM. In *Document Analysis and Recognition, International Conference on,* (p. 1123).

Lee, S. & Pan, J. C. (1992). Offline tracing and representation of signatures.

Systems, Man and Cybernetics, IEEE Transactions on, *22*(4), 755–771.

Levy, N., Wolf, L., Dershowitz, N. & Stokes, P. (2012). Estimating the distinctiveness of graphemes and allographs in palaeographic classification. *DH2012 Book of Abstracts*, 264–267.

Lewis, J. R. & Sauro, J. (2009). The factor structure of the system usability scale. In *Human centered design* (pp. 94–103). Springer.

Li, X., Parizeau, M. & Plamondon, R. (1998). Segmentation and reconstruction of on-line handwritten scripts. *Pattern Recognition*, *31*(6), 675–684.

Lindsay, W. M. (1914). The new palaeography. *The Classical Review*, *28*(06), 209–210.

Loew, E. A. (1914). *The beneventan script: a History of the South Italian Minuscule*.

Long Jr, A. C., Landay, J. A., Rowe, L. A. & Michiels, J. (2000). Visual similarity of pen gestures. In *SIGCHI conference on Human Factors in Computing Systems, Proceedings of the* (pp. 360–367).

Lydon, K. S. (2015). Designing devanagari type: the effect of technological restrictions on current practice.

Maarse, F. J. & Thomassen, A. J. (1983). Produced and perceived writing slant: Difference between up and down strokes. *Acta psychologica*, *54*(1), 131–147.

Mačutek, J. (2008). Runes: Complexity and distinctivity. *Glottometrics*, *16*.

Mallon, J. (1937). Le problème de l'évolution de la lettre. *Arts et Métiers graphiques*, *59*, 25–30.

Mallon, J. (1952). *Paléographie Romaine* (Vol. 3). Consejo Superior de Investigaciones Científicas, Instituto Antonio de Nebrija, de Filología.

Mallon, J., Marichal, R. & Perrat, C. (1939). L'écriture latine de la capitale romaine à la minuscule.

Meuthen, E. & Prevenier, W. (1968). *De oorkonden der graven van Vlaanderen (1191-aanvang 1206).* JSTOR.

Mitra, S. & Acharya, T. (2007). Gesture recognition: A survey. *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, *37*(3), 311–324.

Morasso, P. (1981). Spatial control of arm movements. *Experimental brain research*, *42*(2), 223–227.

Morasso, P. & Ivaldi, F. M. (1982). Trajectory formation and handwriting: a computational model. *Biological cybernetics*, *45*(2), 131–142.

Müller, M. (2007). Dynamic time warping. *Information retrieval for music and motion*, 69–84.

Nel, E.-M., du Preez, J. A. & Herbst, B. M. (2005). Estimating the pen trajectories of static signatures using hidden Markov models. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, *27*(11), 1733–1746.

Nguyen, V. & Blumenstein, M. (2010). Techniques for static handwriting trajectory recovery: a survey. In *9th IAPR International Workshop on Document Analysis Systems, Proceedings of the* (pp. 463–470).

Niels, R. & Vuurpijl, L. (2005). Using Dynamic Time Warping for intuitive handwriting recognition. In *International Graphonomics Society, Proceedings of the 12th* (pp. 217–221).

Ojha, G. (1918). Bhāratīya pracīna lipimālā. *Munshiram Manoharlal, Delhi, India*.

Ore, O. & Ore, Y. (1962). *Theory of graphs* (Vol. 38). American Mathematical Society Providence, RI.

O'Reilly, C. & Plamondon, R. (2007). A software assistant for the design and analysis of neuromuscular tests. In *IEEE Biomedical Circuits and Systems conference (BIOCAS) 2007* (pp. 107–110).

O'Reilly, C. & Plamondon, R. (2009a). Development of a Sigma–Lognormal representation for on-line signatures. *Pattern Recognition*, *42*(12), 3324–3337.

O'Reilly, C. & Plamondon, R. (2009b). A lognormal framework for human movement rehabilitation. *Rehabilitation engineering*, 157–172.

Papaodysseus, C., Rousopoulos, P., Arabadjis, D., Panopoulou, F. & Panagopoulos, M. (2010). Handwriting automatic classification: Application to ancient Greek inscriptions. In *Autonomous and Intelligent Systems (AIS), 2010 International Conference on* (pp. 1–6).

Peignot, C. (1937). Le peignot. *Arts et Métiers graphiques*, *59*, 25–30.

Peust, C. (2006). *Script complexity revisited.* Glottometrics.

Plamondon, R. (1995). A kinematic theory of rapid human movements.

part I: Movement representation and generation. *Biological cybernetics*, *72*(4), 295–307.

Plamondon, R. & Djioua, M. (2006). A multi-level representation paradigm for handwriting stroke generation. *Human movement science*, *25*(4), 586–607.

Plamondon, R. & Maarse, F. J. (1989). An evaluation of motor models of handwriting. *Systems, Man and Cybernetics, IEEE Transactions on*, *19*(5), 1060–1072.

pointypo. (2013). *Jean Mallon paléographe, l'histoire de l'écriture en mouvement.* `http://www.pointypo.com/jean-mallon-paleographe-lhistoire-de-lecriture-en-mouvement/`. (Online; accessed 19-April-2016)

Poulle, E. (1974). Paléographie et méthodologie: Vers l'analyse scientifique des écritures médiévales. *Bibliothèque de l'École des chartes*, 101–110.

Pratesi, A. (1998). Commentare Bischoff: un secondo intervento. *Scrittura e Civiltà*(22), 405–408.

Ramaiah, C., Plamondonm, R. & Govindaraju, V. (2014). A Sigma-Lognormal model for handwritten text CAPTCHA generation. In *Pattern Recognition (ICPR), 22nd International Conference on* (pp. 250–255).

Rubine, D. (1991). Specifying Gestures by example. *Computer Graphics*, *25*(4).

Rumble, A. R. (1994). Using Anglo-Saxon manuscripts. *Anglo-Saxon manuscripts: basic readings*, 3–24.

Salomon, R. (1998). *Indian epigraphy: a guide to the study of inscriptions in Sanskrit, Prakrit, and the other Indo-Aryan languages*. Oxford University Press.

Salomon, R. (2012). Some principles and patterns of script change. *The shape of script: How and why writing systems change*, 119–133.

Sculley, D. & Pasanek, B. M. (2008). Meaning and mining: the impact of implicit assumptions in data mining for the humanities. *Literary and Linguistic Computing*, *23*(4), 409–424.

Serva, M. & Petroni, F. (2008). Indo-European languages tree by Levenshtein distance. *EPL (Europhysics Letters)*, *81*(6), 68005.

Singer, Y. & Tishby, N. (1994). Dynamical encoding of cursive handwriting. *Biological Cybernetics*, *71*(3), 227–237.

Soumya, A. & Kumar, G. H. (2014). Classification of Ancient Epigraphs into different periods using Random Forests. In *Signal and Image Processing (ICSIP), 5th International Conference on* (pp. 171–178).

Stansbury, M. (2009). The computer and the classification of script. *Kodikologie und Paläographie im Digitalen Zeitalter–Codicology and Palaeography in the Digital Age*, 237–249.

Stokes, P. (2007). Palaeography and image-processing: some solutions and problems. *Digital Medievalist*, *3*, 2007–20.

Stokes, P. (2009a). Computer-aided palaeography, present and future. *Kodikologie und Paläographie im Digitalen Zeitalter–Codicology and Palaeography in the Digital Age*, 309–338.

Stokes, P. (2009b). *Hand Analyzer.* `http://sourceforge.net/projects/handanalyser/`. (Accessed: 2016-03-15)

Stokes, P. (2012). Modeling medieval handwriting: A new approach to digital palaeography. *DH2012 Book of Abstracts, Digital Humanities 2012*.

Stokes, P., Brookes, S., Noel, G., Buomprisco, G., Marques De Matos, D. & Watson, M. (2014). The Digipal Framework for Script and Image. In *DH2014 Book of Abstracts, Digital Humanities 2014.* University of Lausanne.

Terras, M. & Robertson, P. (2004). Downs and acrosses: textual markup on a stroke level. *Literary and linguistic computing*, *19*(3), 397–414.

Teulings, H.-L. & Schomaker, L. R. (1993). Invariant properties between stroke features in handwriting. *Acta psychologica*, *82*(1), 69–88.

Tucha, O., Tucha, L. & Lange, K. W. (2008). Graphonomics, automaticity and handwriting assessment. *Literacy*, *42*(3), 145–155.

Vatavu, R.-D., Vogel, D., Casiez, G. & Grisoni, L. (2011). Estimating the perceived difficulty of pen gestures. In *Human-Computer Interaction–INTERACT 2011* (pp. 89–106). Springer.

Wallace, C. S. & Dowe, D. L. (1999). Minimum message length and Kolmogorov complexity. *The Computer Journal*, *42*(4), 270–283.

Wang, J., Wu, C., Xu, Y.-Q. & Shum, H.-Y. (2005). Combining shape and physical models for online cursive handwriting synthesis. *In-*

ternational Journal of Document Analysis and Recognition (IJDAR), *7*(4), 219–227.

Wang, J., Wu, C., Xu, Y.-Q., Shum, H.-Y. & Ji, L. (2002). Learning-based cursive handwriting synthesis. In *Frontiers in Handwriting Recognition, 8th International Workshop on* (pp. 157–162).

Willems, D. & Niels, R. (2008). Definitions for features used in online pen gesture recognition. *Nijmegen Institute for Cognition and Information Radboud University Nijmegen, Nijmegen, The Netherlands*.

Willems, D., Niels, R., van Gerven, M. & Vuurpijl, L. (2009). Iconic and multi-stroke gesture recognition. *Pattern Recognition*, *42*(12), 3303–3312.

Wolf, L., Potikha, L., Dershowitz, N., Shweka, R. & Choueka, Y. (2011). Computerized paleography: tools for historical manuscripts. In *Image Processing (ICIP) 2011, 18th IEEE International Conference on* (pp. 3545–3548).

Wright, D. B. & Williams, S. E. (2005). Three dimensional (3D) scatterplots. *Encyclopedia of statistics in behavioral science*.

Xu, S., Lau, F., Cheung, W. K. & Pan, Y. (2005). Automatic generation of artistic Chinese calligraphy. *Intelligent Systems, IEEE*, *20*(3), 32–39.

List of Tables

List of Figures