Open Ended Lab Project Report
TRACING SCRIPT EVOLUTION THROUGH ANALYSIS OF HANDWRITTEN KANNADA TEXTS

Mentor: Dr. Vishwas V
Students: Mausumi Bhuyan (122201001), Anusha Dadam (112201043)

1 Introduction

Indic scripts, such as Kannada and Malayalam, carry profound cultural and linguistic significance, embodying centuries of heritage and tradition. However, despite their rich history, handwritten versions of these scripts present a persistent challenge for modern technological applications like Optical Character Recognition (OCR) and Machine Learning (ML). The complexity of these scripts, with their intricate ligatures and diverse writing styles, often defies conventional recognition methods.

The early evolution of writing systems can be traced back to the Indus Valley Civilization, where some of the earliest forms of writing were discovered. The Brahmi Script, which emerged in the 3rd century BCE, is considered the forerunner to all Indian scripts. By deciphering these ancient scripts, we gain valuable insights into the development and transformation of writing systems over time.

Brahmi script is the oldest deciphered script in India, known for Ashoka's rock-cut edicts. It's an abugida written left to right, meaning each character represents a consonant with an inherent vowel sound typically 'a'.

In the ancient Harappan civilization, they used a special kind of writing made up of pictures and symbols, totaling 396 different ones, unlike modern writing that goes from left to right, their writing went from right to left. Indus/Harappan script is a pictographic which means all the character in the script resembles the pictures of the real world.

Here we can see that both Malayalam and Kannada languages are derived from the Brahmi script and these are more circular when compared to Brahmi and both languages are 3rd generation languages in the above chart.

Recognizing the urgency of this challenge, our report endeavors to shed light on the evolution of handwritten Kannada and Malayalam scripts.

1.1 Objectives

There are two main objectives for this project:

- Implementing and uniquely identifying Kannada and Malayalam Scripts.
- Characterizing the variations in writing styles and tracing the evolution of writing methods over time.

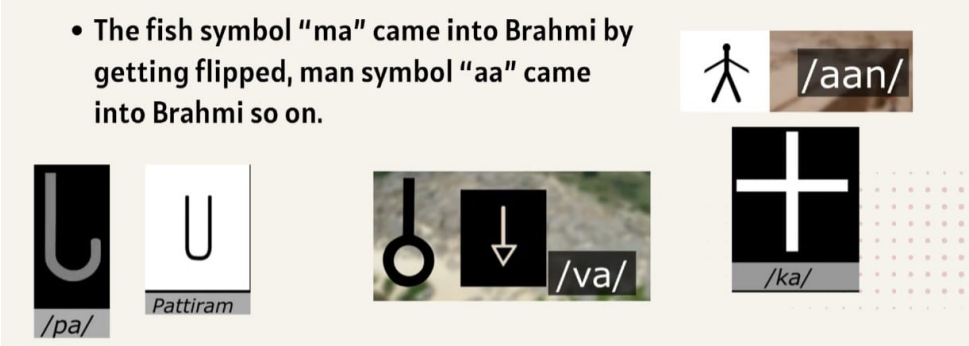

Figure 1: Example Image

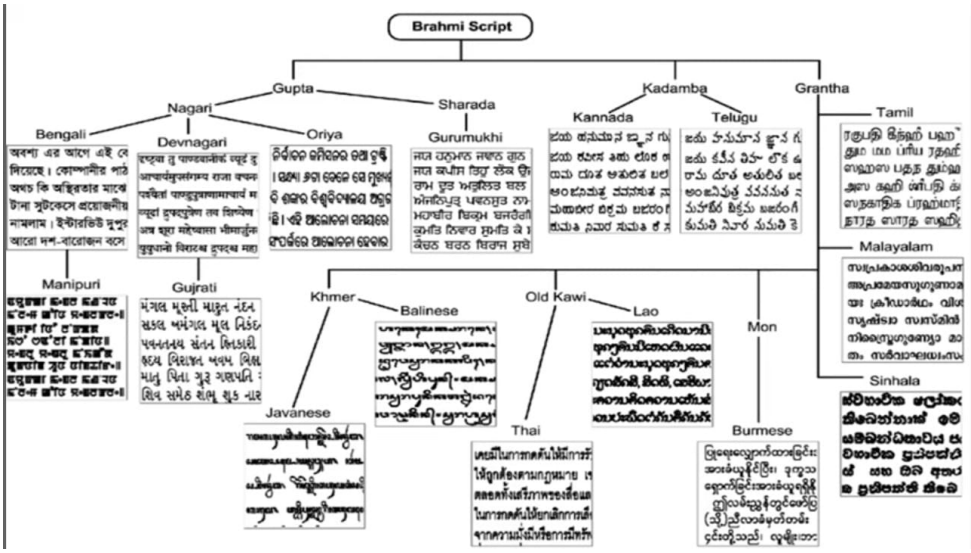

Figure 2: Example Image

2 Characteristics of Scripts

- Letter composition in these languages is considerably more complicated. The importance of quantification arises from the complexity of languages. Unlike English, Kannada scripts contain nearly 15 vowels & 34 consonants and many compound letters using these.
- Quantifying scripts is essential for extracting meaningful insights from textual data in a structured format, empowering comprehensive analysis and driving progress across diverse domains reliant on textual data.

2.1 Visual Information

This refers to the appearance of the character such as length, average area, average curvature and compactness.

In our approach to quantification, we focus on the following crucial aspects:

- Length
- Area
- Size
- Compactness
- Average Curvature
- Circularity

3 Methodology

3.1 Order Parameter

- Pixel Density Correlation:2022 OELP

Pixel Density Correlation, utilized in the previous OELP and effective for English, proved less effective for Kannada and Malayalam characters due to subtle distinctions. As we can see the radial distribution curves are almost the same for two distinct Malayalam and Kannada letters.

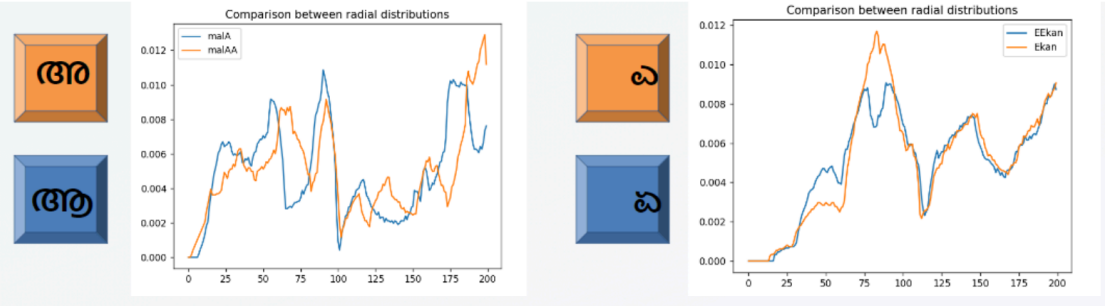

Figure 3: Example Image

- Circular Hough Transform

  The Circular Hough Transform used in previous oelp(2023) is an image processing technique used to find circles within an image. The algorithm begins by detecting edges in the input image, often achieved through an edge detection method like the Canny edge detector. Next, it builds a parameter space representing potential circles, considering different center coordinates and radii. For each edge point, the algorithm accumulates votes in the parameter space, and peaks in this space indicate potential circles. A threshold helps filter out noise, ensuring only significant circles are detected.

  But even in this circular hough transformation the results aren't good, since different charecters may have same number of circles.

  So , since both pixel density correlation and circular hough transform are not working effectively for kannada and Malayalam scripts, so we came up with the technique based on visual information of a letter.

Figure 4: Example Image

Our key innovation lies in the creation of a unique numerical representation known as the order parameter for each script character. This order parameter incorporates a holistic analysis of various character attributes such as length, area, size, curvature, circularity, and compactness.

This order parameter isn't just a simple count of strokes or loops; rather, it incorporates a holistic analysis of various character attributes such as length, area, size, curvature,circularity and compactness. By quantifying these characteristics, we create a nuanced framework that enables precise discrimination between script characters, even amidst variations in handwriting styles.

3.2 Metrics

The following metrics attempt to quantify the different aspects of visual information:

- Length: Length is defined as perimeter of a character, which is the sum of lengths of the contour, contour is defined as a curve joining all the continuous points along the boundary having same color or intensity.

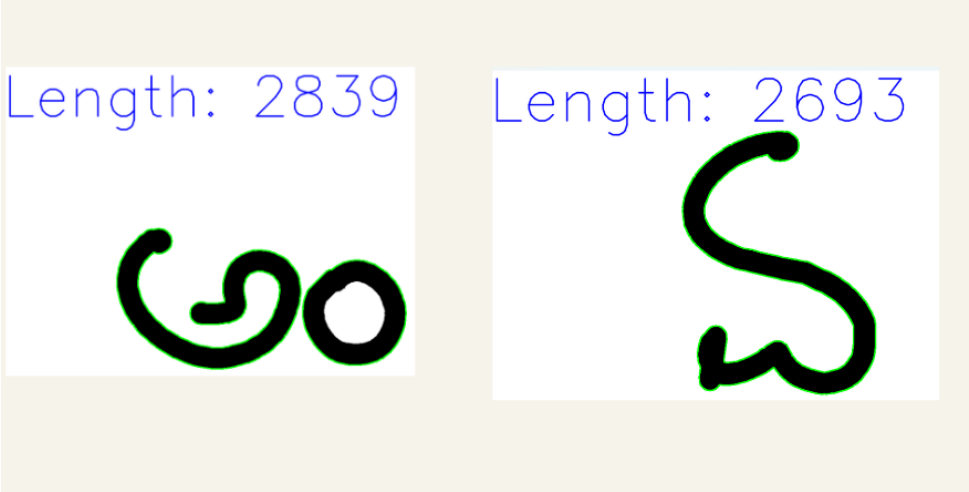

Figure 5: Example Image

Since there are two different letters but the length is almost similar , so for order of parameter we can not take only length. So the need for other visual informations comes in to picture.

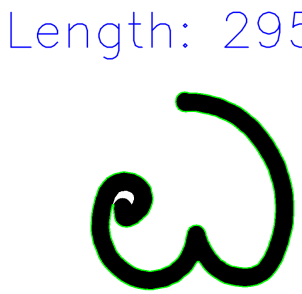

Figure 6: Example Image

- Area: Area is the number of pixels that are present in the region, here it is the number of black pixels present in the input image.

Figure 7: Example Image

- Size: Size is measured by the bounding box area of a character. The bounding box is the minimal rectangle that encloses the character.

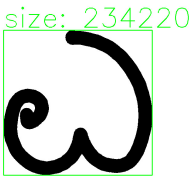

Figure 8: Example Image

- Compactness: refers to how closely packed the pixels are within a character or a text region, used to differentiate letters based on their density. It defines how compact a character appears, it is defined as length/size here.

  Ex1:

Character Length: 2960

Bounding box coordinates: (173, 277, 704, 364)

Size: 256256

comp: 0.0115

Ex2:

Character Length: 2959

Bounding box coordinates: (229, 208, 490, 478)

Size: 234220

comp: 0.0126

- Average Curvature: a measure of how much a curve deviates from being straight. It is the average of all key points curvature, a key point is a point in the character where there is a significant change in the shape of a character.

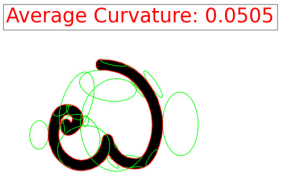

Figure 9: Example Image

- Circularity: It is defined as $(\text{length})^2/4 * pi * area$

Ex:

Character Length: 2960

circularity is : 7.5776

4 Results

Imperfect Vraiations 1	Length	Size	Compactness	Curvature
a	2550	115328	0.022	0.0111
ta	1597	150480	0.011	0.0596
eh	1419	62744	0.023	0.0155
na	2268	155040	0.015	0.0005
o	1680	107623	0.016	0.067

Figure 10: Example Image

Imperfect Variations 2	Length	Size	Compactness	Curvature
a	2644	142896	0.018	0.0039
ta	1398	117623	0.012	0.0514
eh	2749	183997	0.015	0.0104
na	3561	366507	0.009	0.0081
o	2215	153375	0.014	0.0039

Figure 11: Example Image

Imperfect Variation 3	Length	Size	Compactness	Curvature
a	2862	175310	0.016	0.0035
ta	1641	167670	0.009	0.0614
eh	2691	202986	0.013	0.0022
na	2772	196578	0.014	0.0004
o	3212	263442	0.012	0.0039

Figure 12: Example Image

Perfect Variation	Length	Size	Compactness	Curvature
a	2483	124690	0.019	0.0437
ta	1632	166112	0.009	0.0554
eh	2175	124690	0.017	0.0321
na	2721	171402	0.015	0.036
o	2225	144078	0.015	0.051

Figure 13: Example Image

4.1 Correlation Between Matrices

We took 5 letters and found the correlation between them. For each perfect letter, 5 variations have been taken and done the analysis.

1. Length vs Size: if we denote length as “L” then size would be almost “ L*L ”. The order of graph follows like ta,eh,o,na,a.

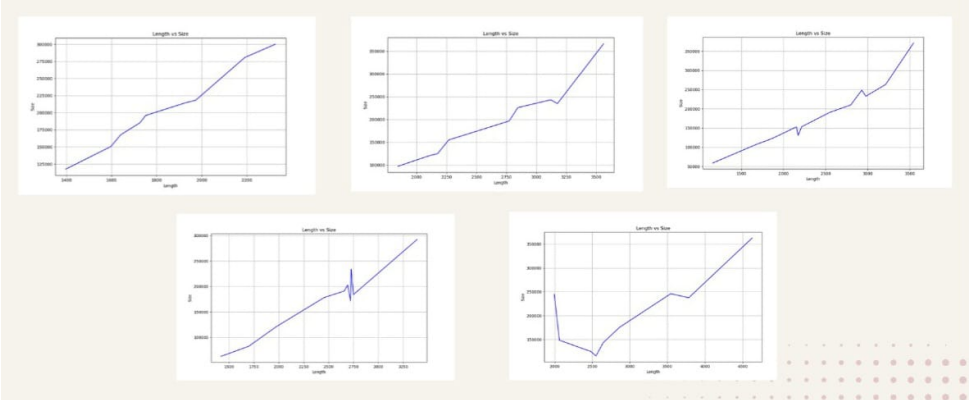

Figure 14: Example Image

2. Length vs Compactness: length would be “L”, compactness would be “1/L”, from the definition of compactness mentioned above. The order of graph follows like ta,eh,o,na,a.

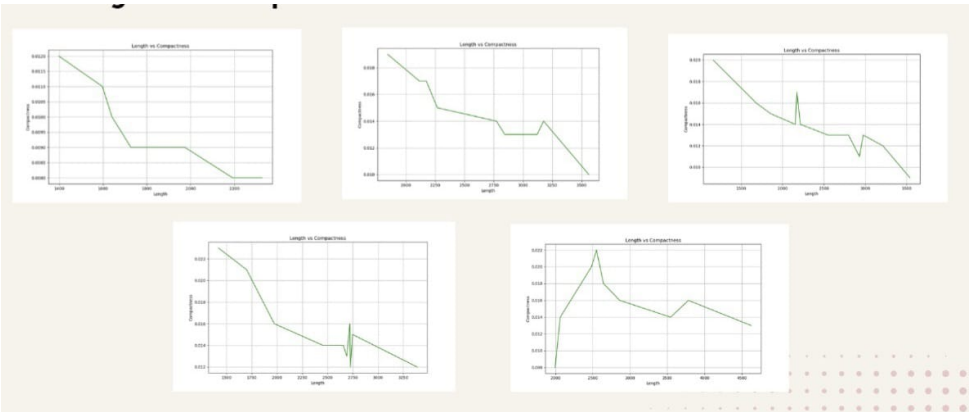

Figure 15: Example Image

3. Size vs Compactness: Size is “L*L”, compactness is “1/L”. The order of graph follows like ta,eh,o,na,a.

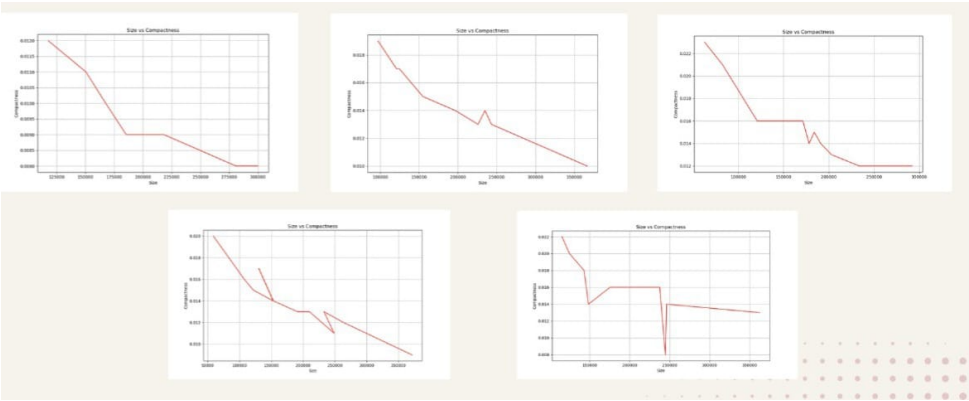

Figure 16: Example Image

4. Average Curvature vs Compactness : The order of graphs is eh,ta,a,na,o.

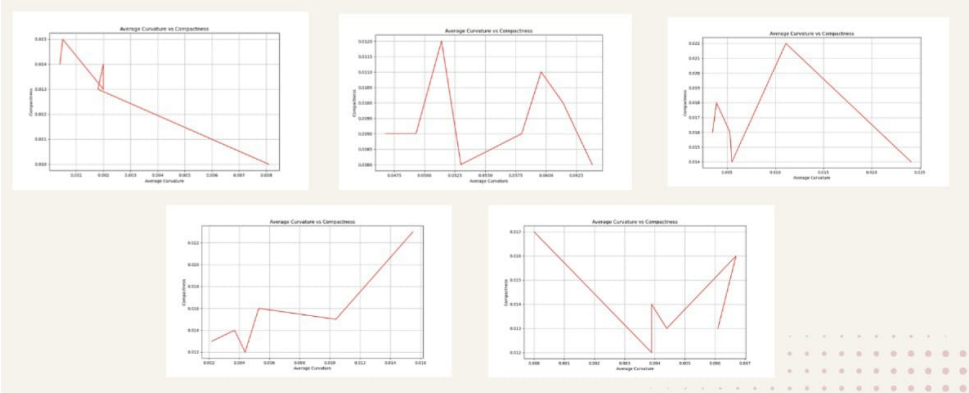

Figure 17: Example Image

4.2 Distribution of the Matrices

For the distribution of length, size, average curvature, and compactness, we took a perfect letter and 50 variations of it.

Here also we did for 5 letters for which correlation have been done.

1. Length

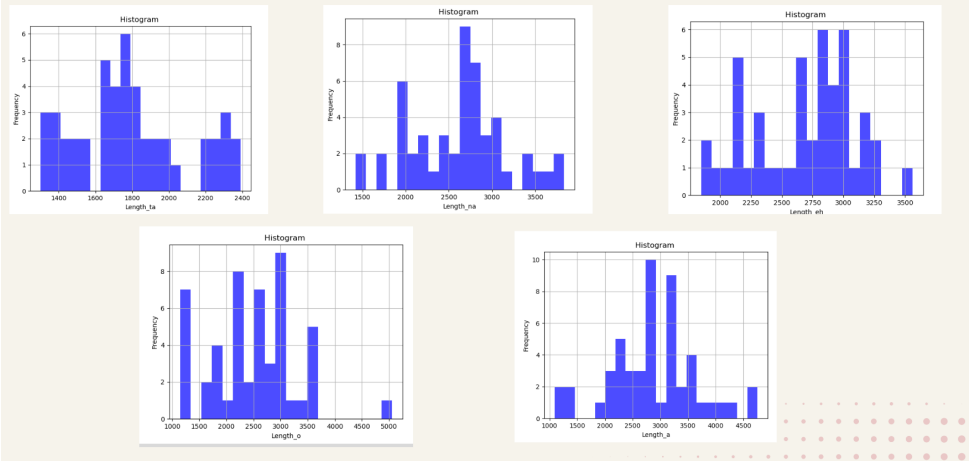

Figure 18: Example Image

2. Size

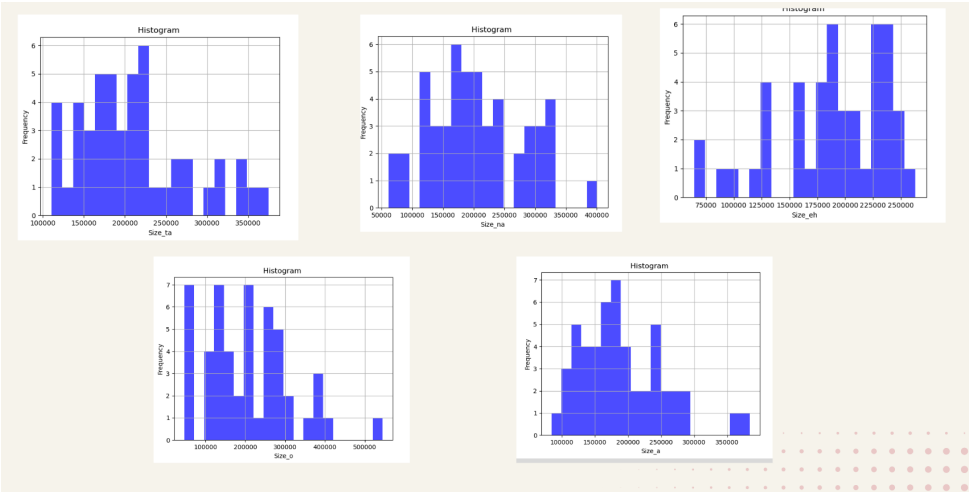

Figure 19: Example Image

3. Compactness

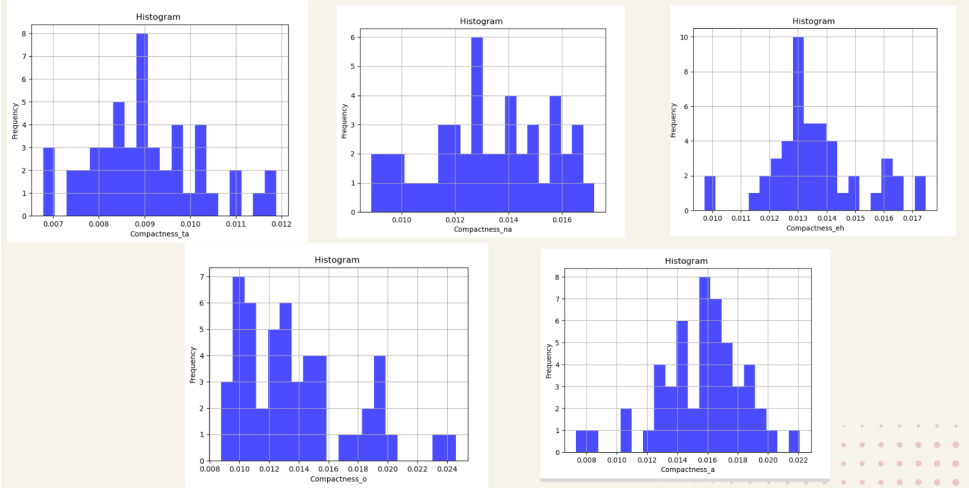

Figure 20: Example Image

4. Average Curvature

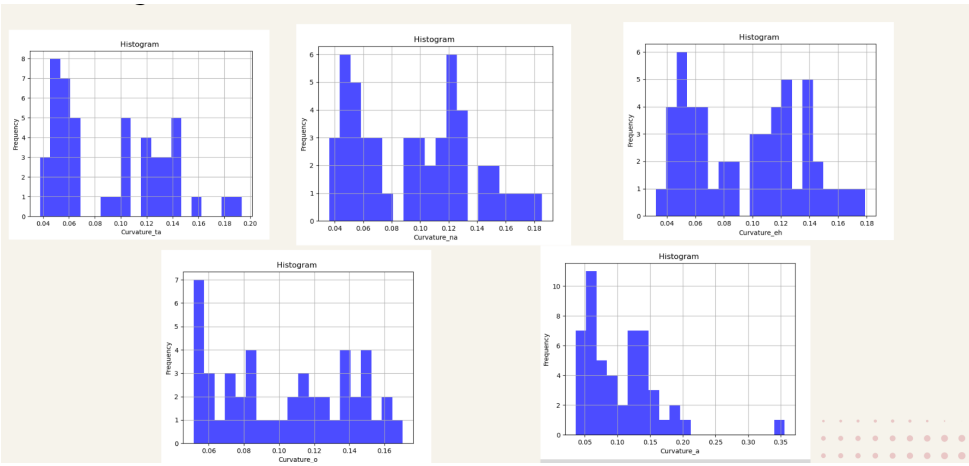

Figure 21: Example Image

5 Creating Synthetic Data Sets

Furthermore, we recognize the critical importance of data in training OCR and ML models effectively. Given the limited availability of handwritten script datasets, especially for less commonly studied languages like Kannada and Malayalam, we're taking a proactive approach. In addition to utilizing existing datasets, we're generating synthetic data by introducing controlled variations and noise into our samples.

This approach not only augments the quantity of available data but also enriches the diversity of our training set, enhancing the robustness and adaptability of our recognition algorithms.

Initially random noise, rotation, compression and expansion to the letter have been done to create synthetic data, but with this we cannot generate a significant number of varitions . For this we adopted

the approach Randomised Stroke Variation.

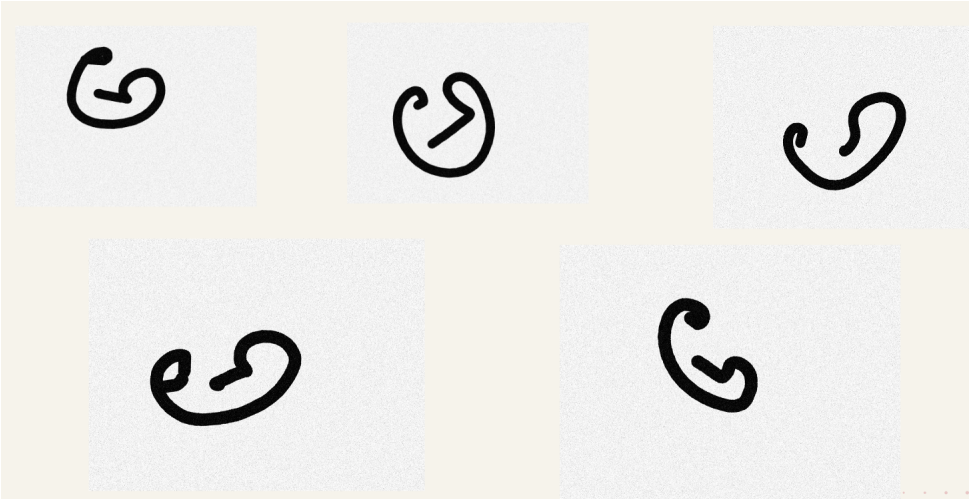

Figure 22: Example Image

5.1 Randomized Stroke Variation

This introduce slight variations in the thickness, curvature, and endpoints of the strokes in the letter. You can achieve this by randomly modifying the paths of the strokes while maintaining the overall shape of the letter.

6 Recognition of Letters Over a Period of Time

Since kannada and Malayalam charecters are derived from brahmi script, if we compare them then the import criteria to decide the timeline to be the circularity.

Where circularity is (length*length)/(4*pi*area).

7 Future Work

- Creating synthetic data using Randomized Stroke Variation.
- Compiling the script from at least 4 different eras and their quantification.

8 Conclusion

we dove into understanding how handwritten Kannada and Malayalam scripts evolved over time. We aimed to identify these scripts uniquely and track how writing styles changed. We tried traditional methods like pixel density correlation and circular Hough transform but found they didn't work well for Kannada and Malayalam. So, we came up with a new idea: using visual details like length, area, curvature, and more to create a special number for each letter. This helped us tell the letters apart even with different handwriting styles. Since there aren't many handwritten scripts available for training

computers, we made our own by tweaking letters slightly. This will help make computer programs better at reading these scripts. In the future, we plan to refine this process and study scripts from different time periods. Our goal is to not only understand how these scripts changed but also make it easier for computers to read them accurately.

9 References

1. Kaggle: https://www.kaggle.com/competitions/Kannada-MNIST/data
2. ResearchGate: https://www.researchgate.net/profile/Chandrashekar-Gudada/publication/349895833$_{H}istorical_{K}a$
$Nearest_{N}eighbour_{T}echnique/links/60464652a6fdcc9c782175a8/Historical-Kannada-Handwritten-$
$Character-Recognition-using-K-Nearest-Neighbour-Technique.pdf$
3. Springer: https://link.springer.com/chapter/10.1007/978-981-19-1844-5$_{5}$6
4. IEEE Xplore: https://ieeexplore.ieee.org/abstract/document/7019645
5. Proceedings of the National Academy of Sciences: https://www.pnas.org/doi/abs/10.1073/pnas.0906237106
6. JetIR: https://www.jetir.org/papers/JETIR2004600.pdf
7. Science: https://www.science.org/doi/10.1126/science.1170391
8. GitHub: github.com/VikParuchuri/s
9. YouTube: https://youtu.be/N$_{t}r9IoB_{h}k?feature=shared$
10. Tenso: https://www.tensoic.com/blog/kannada-llama/?s