FROM WORDS TO STROKES

A PROJECT REPORT

Submitted by

Sapeksh Pareek (211B274)
Sarthak Nagar (211B276)
Shivansh Bhatnagar (211B296)

Under the guidance of: Dr. Rahul Pachauri

Dec - 2023

Submitted in partial fulfillment for the award of the degree of

BACHELOR OF TECHNOLOGY

IN

COMPUTER SCIENCE & ENGINEERING

Department of Computer Science & Engineering
JAYPEE UNIVERSITY OF ENGINEERING & TECHNOLOGY
AB ROAD, RAGHOGARH, DT. GUNA-473226 MP, INDIA

Declaration by the Students

We hereby declare that the work reported in the B. Tech. project entitled as "**FROM WORDS TO STROKES**", in partial fulfillment for the award of degree of Bachelor of Technology submitted at **Jaypee University of Engineering and Technology, Guna**, as per best of our knowledge and belief there is no infringement of intellectual property right and copyright. In case of any violation, we will solely be responsible.

Sapeksh Pareek (211B274)

Sarthak Nagar (211B276)

Shivansh Bhatnagar (211B296)

Department of Computer Science and Engineering

Jaypee University of Engineering and Technology

Guna, M.P., India

Date: 27/11/2023

JAYPEE UNIVERSITY OF ENGINEERING &TECHNOLOGY
Grade 'A+' Accredited with by NAAC & Approved U/S 2(f) of the UGC Act, 1956
A.B. Road, Raghogarh, Dist.: Guna (M.P.) India, Pin-473226
Phone: 07544 267051, 267310-14, Fax: 07544 267011
Website: www.juet.ac.in

CERTIFICATE

This is to certify that the work titled "**FROM WORDS TO STROKES**" submitted by "**Sapeksh Pareek (211B274), Sarthak Nagar (211B276), Shivansh Bhatnagar (211B296)**" in partial fulfillment for the award of degree of B.Tech of **Jaypee University of Engineering & Technology, Guna** has been carried out under my supervision. As per best of my knowledge and belief there is no infringement of intellectual property right and copyright. Also, this work has not been submitted partially or wholly to any other University or Institute for the award of this or any other degree or diploma. In case of any violation concern student will solely be responsible.

Signature of Supervisor

Dr. Rahul Pachauri

Assistant Professor (SG)

Date: 27/11/2023

ACKNOWLEDGEMENT

First and foremost, we would like to express our gratitude to Jaypee University of Engineering and Technology, our college, for giving us this chance to share our collective talents. We would also like to extend our sincere gratitude to Dr. Rahul Pachauri, our project mentor and supervisor, who has supported us during all the challenges and issues we encountered enroute to finishing our project. We also want to express our gratitude to our mentor, who spent valuable time proofreading and fixing our mistakes in addition to offering us solutions and pointing us in the right direction. We also acknowledge the invaluable assistance provided by our parents and friends in helping us to complete this project on time. To cap it off, we extend our heartfelt thanks to the team members of "FROM WORDS TO STOKES."

Sapeksh Pareek (211B274)
Sarthak Nagar (211B276)
Shivansh Bhatnagar (211B296)

Date: 27/11/2023

EXECUTIVE SUMMARY

Our project, centered around the development of distinctive shorthand symbolic fonts named 'Rishi Pranali,' represents a significant leap in enhancing the visual identity and efficiency of the Indian Judiciary system. Using Illustrator, we meticulously crafted these fonts, ensuring seamless integration into our system for a unique project identity.

To overcome the challenges of recognizing Hindi characters within the shorthand, we harnessed the capabilities of an existing OCR system. This strategic integration facilitated the successful identification of Hindi characters, laying the foundation for efficient transcription in the Indian judicial context.

Recognizing the absence of a dedicated dataset for our shorthand 'Rishi Pranali' symbolic language, we took the initiative to create a bespoke dataset. Leveraging the LeNet-5 architecture and Python 3.9, we designed algorithms tailored to the intricacies of our shorthand, ensuring the dataset's relevance and accuracy.

The development of this unique dataset serves as a critical component in our mission to accelerate and streamline the Indian Judiciary system. By harnessing the power of advanced technologies and customized algorithms, we aim to significantly enhance the processing speed and accuracy of shorthand utterance transcription.

Our project aligns with the broader goal of modernizing transcription processes within the Indian judicial framework. The integration of distinctive shorthand fonts, coupled with a purpose-built dataset and advanced OCR technology, positions our initiative at the forefront of innovation in legal transcription. As we work towards making the Rishi Pranali shorthand a cornerstone in the judicial communication landscape, our project stands as a testament to the fusion of design, technology, and efficiency in service of a crucial societal institution.

Table of Figures

Figure 1.1: Rishi Pranali 13
Figure 3.1: System Design 20
Figure 5.1: Success Rate 27
Figure 5.2: Lose Rate 27
Figure 5.3: Output 27

Table of Contents

Title Page i
Declaration by the Students ii
CERTIFICATE iii
ACKNOWLEDGEMENT iv
Table of Figures vi
Table of Contents vi
Chapter 1 - Introduction 10
1.1 Problem Statement 10
1.2 Motivation for Work 10
1.3 Project Overview 11
1.3.1 What is Shorthand 11
1.3.1.1 Types of Shorthand 11
1.3.2 Court Usage 12
1.3.2.1 Unified Symbolism 12
1.3.3 International 12
1.3.3.1 Gregg 12
1.3.3.2 Pitman 13
1.3.4 National 13
1.3.2.1 Rishi Pranali 13
1.4 System Requirements 14
1.4.1 Hardware Requirements 14
1.4.2 Software Requirements 14
Chapter 2 – Literature Survey 15
2.1 Existing Model 15
2.1.1 Gregg Model 15
2.2 Proposed Model 16
Chapter 3 – System Analysis & Design 16

3.1 Requirement Specifications 17
3.1.1 Dataset Used 17
3.1.2 Python 17
3.1.3 Python Libraries 18
3.1.4 PyCharm 19
3.1.5 Jupyter Notebook 19
3.1.6 Illustrator 19
3.1.7 Photoshop 20
3.1.8 Caliber 20
3.2 System Design 20
3.2.1 System Architecture 20
3.2.2 OCR Architecture 21
3.3 Methodology/Process 21
Chapter 4 – Feature and Extraction 24
Chapter 5 – Result and Discussion 25
Chapter 6 – Conclusion 28
Chapter 7 – Referenecs 29
Chapter 8 – Contributers 30

Chapter 1

Introduction

1.1 Problem Statement

The inefficiency in transcribing courtroom proceedings in the Indian Judiciary system prompts the need for a specialized solution. The absence of dedicated shorthand symbolic fonts and a tailored dataset for accurate transcription of Hindi characters poses a significant challenge, leading to a slow and error-prone process. Existing Optical Character Recognition (OCR) systems struggle to comprehend the intricacies of the unique 'Rishi Pranali' shorthand language. This project aims to address these issues by developing distinct shorthand fonts, creating a bespoke dataset, and enhancing OCR capabilities. The primary goal is to expedite and streamline transcription processes, ensuring a faster and more accurate documentation of courtroom utterances in the Indian legal framework. Ultimately, this initiative seeks to increase work efficiency for stenographers in court and alleviate the workload burden within the judiciary.

1.2 Motivation for Work

Our project is an example of our steadfast dedication to quality and our proficiency in the rapidly evolving field of legal transcription. We are setting out to transform this field by bringing creativity to bear by producing a unique "Rishi Pranali" Shorthand Dataset. This project greatly advances the modernization of legal transcription procedures while also demonstrating our technological prowess. Through the integration of state-of-the-art technologies and proprietary algorithms, our goal is to completely redefine industry standards. Furthermore, our project aims to lighten the load on the judiciary, which is a noble goal. We work to ensure timely and accurate documentation of court proceedings by putting in place an effective transcription system, which improves the judicial workflow. We are committed to using technological innovation to advance societal progress, and as we take on this challenge, our efforts align with the larger goals of the Digital India Initiative. This project is a brave step towards influencing the direction of legal transcription and aiding in India's digital revolution, not merely a showcase of capabilities.

1.3 Project Overview

We have developed unique 'Rishi Pranali' shorthand fonts using Illustrator, seamlessly integrating them for a distinct project identity. Leveraging OCR capabilities, we successfully recognized Hindi characters and created a bespoke dataset. Our initiative aims to expedite Indian Judiciary transcription by enhancing processing speed and accuracy, marking a significant stride toward modernizing legal transcription processes.

1.3.1 What is Shorthand

Shorthand is an organized approach to writing quickly that uses special characters, abbreviations, and symbols to represent spoken words or sounds. Its main function is to make taking notes faster and more effective, which makes it invaluable in situations when recording spoken information in real time is essential. There are numerous shorthand systems, each with its own set of symbols and conventions. Pitman Shorthand is well known for using hand positions and stroke directions to indicate sounds. It was created by Sir Isaac Pitman. John Robert Gregg developed Gregg Shorthand, which uses elliptical figures and lines for efficiency. A more recent version, Teeline Shorthand, makes learning easier by following a few fundamental guidelines. Forkner Shorthand incorporates cursive characteristics, while other systems, such as Speedwriting, concentrate on shortening complete words. Applications for shorthand can be found in secretarial work, court reporting, and journalism. Its continued significance stems from its quick information collection speed, which enables users to follow spoken dialogue. Shorthand is still a useful talent in our fast-paced world of communication because it allows people to copy spoken words quickly and accurately, which boosts productivity in a variety of professional and academic contexts.

1.3.1.1 Types of Shorthand

Shorthand can be divided into two main categories:

Text-based: A typical shorthand system provides symbols or abbreviations for words and common phrases, which can allow someone well-trained in the system to write as quickly as people speak.

Phonetic Based: The words are mostly written as they are spoken, with the symbols representing sounds rather than letters.

1.3.2 Court Usage

For the precise recording of spoken words during court hearings, shorthand is essential in legal situations. With their proficiency in shorthand, stenographers are essential to this procedure. They record verbal conversations, witness statements, and meetings in real time using stenotype keyboards or specialized shorthand devices. The ability to create official transcripts is made possible by the stenographer's expertise in shorthand, which is of great assistance to legal specialists. This thorough documentation speeds up the transcription process and makes thorough reference in later legal actions possible by guaranteeing an exact record of courtroom activities.

1.3.2.1 Unified Symbolism

Shorthand guarantees the dependability and correctness of court records in both domestic and international settings, which is essential for legal procedures, appeals, and historical recordkeeping. The fundamentals of shorthand are still important for court reporting, even if some court reporters are using stenography machines or other digital tools to help with the transcription process as technology progresses.

1.3.3 International

Over linguistic barriers, international shorthand provides a standardized and effective way to transcribe spoken language. International shorthand is used to take notes quickly and accurately in a variety of professional settings, including business meetings, legal proceedings, and journalism. It is a useful tool for efficiently and globally applicable recording of spoken communication due to its versatility in handling different languages and its succinct symbols, which promote efficient communication and documentation. There are two types of usage of International Shorthand.

1.3.3.1 Gregg

Overview: John Robert Gregg invented Gregg Shorthand, which is renowned for its quickness and ease of use. It is effective for taking notes quickly because it represents sounds with elliptical figures, lines, and curves.

International Application: Internationally, especially in English-speaking nations, Gregg Shorthand is widely used. Its adaptability and simple symbols have made it popular in a variety of contexts, such as business meetings, courtrooms, and educational institutions.

Light line and Heavy-line versions: There are light-line and heavy-line versions available from Gregg Shorthand. Although the heavy-line version is quicker, writing accuracy is higher. The decision between the two frequently comes down to personal taste and the needs of the user.

1.3.3.2 Pitman

Overview: One of the oldest and most popular shorthand systems is Pitman Shorthand, created by Sir Isaac Pitman. It is dependent on the hand's position and stroke direction and uses strokes, curves, and dots to represent sounds.

International Application: Pitman Shorthand is widely used in many English-speaking nations and has gained widespread adoption. It became well-known for its methodical approach to sound representation and its adaptability in precisely capturing spoken words.

Variation and Adaptation: Pitman Shorthand has several iterations and adaptations that are tailored to the unique linguistic quirks of various locales. Its adoption and usage on a global scale have been facilitated by these modifications.

1.3.4 National

Shorthand is widely used in court reporting in India, allowing court reporters to accurately and quickly transcribe spoken words. Quick note-taking during interviews or press conferences is also essential in journalism. Shorthand ensures effective documentation in meetings and conferences, which is beneficial for administrative professionals. As a useful skill, shorthand helps professionals in a variety of professional and educational settings by enabling them to quickly capture information, which helps with real-time transcription and concise record-keeping. This boosts productivity in a variety of sectors.

1.3.2.1 Rishi Pranali

In Indian courts, proceedings are conducted using the symbolic language known as Rishi Pranali, which is essential to the functioning of the Indian judiciary. This language is manually transcribed into Hindi .

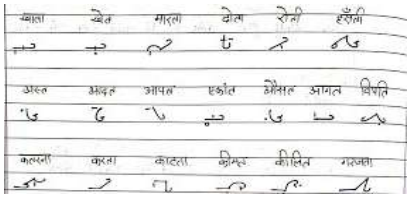

Figure 1.1: Rishi Pranali

and is specifically used by stenographers who are seated in courtrooms with their typewriters. It is an essential tool for precise and effective court hearing documentation.

1.4 System Requirements

The thorough rundown of the system prerequisites necessary for our creative dataset for Rishi Pranali to be implemented and run successfully, guaranteeing a thorough comprehension of the technology infrastructure behind this ground-breaking project.

1.4.1 Hardware Requirements

1. Processor (Recommended):
   Intel Core i5 12th Gen or AMD Ryzen 5 4th Gen or higher.
2. Input (Recommended):
   Touch Screen Laptop, Wacom Tablet with Stylus.
3. Graphics Processing unit (GPU) (Recommended):
   Nvidia GeForce, AMD Radeon 4GB or higher.
4. Storage (Recommended):
   256 SSD or higher.
5. Random Access Memory (RAM) (Recommended):
   8 GB DDR$ or higher.

1.4.2 Software Requirements

1. Operating System: Windows 10 or macOS X or higher.
2. Programming Language: Python 3.7 or higher.
3. Integrated Development Environment: PyCharm, Jupyter Notebook
4. Adobe Illustrator v20.0.32 or higher.
5. Adobe Photoshop v22.8.03 or higher.
6. Caliber x64 v7.1
7. Jupyter Notebook v7.1.0a0.

Chapter 2

Literature Survey

2.1 Existing Model

An automated Gregg shorthand word to English word conversion was created using a deep learning methodology. The Inception-v3 Convolutional Neural Network (CNN) model was employed in the TensorFlow platform, an open-source object classification algorithm. With a total of 16,200 datasets, the training datasets comprise 135 legal terminologies and 120 images per word. The validation accuracy of the trained model was 91%. 1,350 handwritten Gregg Shorthand words were tested over the course of 10 trials, one for each legal terminology. With 739 words translated correctly, the system's accuracy was 54.74%.

Many countries have embraced Gregg Shorthand, a widely used shorthand system known for its accuracy in transcription, to take notes quickly and accurately in a variety of situations. Its usefulness is increased by the availability of digital datasets, especially on the "Steno: DEK," especially in court cases that have a tight deadline. The practical utility of this shorthand is demonstrated by a well-known model, credited to the "S.J.Sarman" which makes it easier for court reporters to transcribe quickly. This tried-and-true model guarantees uniformity and functionality across jurisdictions while expediting legal documentation. It is a prime example of Gregg Shorthand's continued global significance in enabling efficient and precise transcribing processes.

Gregg Shorthand is useful for quick transcription, but it has drawbacks as well, like a steep learning curve, potential trouble adjusting to digital platforms, and inconsistent versions that cause problems with standardization. Its main English-language design restricts its application in multilingual environments, and variations in writers' skill levels could affect coherence. Odd words become complicated, and declining demand in contemporary settings with widespread voice recognition technology is concerning. Even though Gregg Shorthand is still useful, particularly for taking notes, its drawbacks highlight the necessity of flexibility and considering new technological options in the ever-changing communication landscape.

2.1.1 Gregg Model

Principle: John Robert Gregg created the phonetic system known as Gregg Shorthand. Instead of using individual letters to represent sounds, it combines curves, lines, and ellipses.

Rather than strictly following spelling rules, it aims to capture spoken words as they sound, operating under the principles of simplicity and speed.

Symbols: Gregg Shorthand uses a range of symbols to represent vowels and consonants. The hand's organic motion serves as the model for the symbols.

Adaptations: Gregg Shorthand comes in a variety of forms, such as Simplified and Anniversary, each with unique enhancements and adjustments to enhance learning and speed.

Model Aspects: In contrast to machine learning models, Gregg Shorthand does not require a model to function. Rather, it's a system of guidelines and notations developed by John Robert Gregg to phonetically represent sounds.

Learning Approach: In order to learn Gregg Shorthand, a person must study its fundamentals, become familiar with the symbols for vowels and consonants, and practice writing words and phrases quickly.

2.2 Proposed Model

As part of a ground-breaking project to improve legal documentation in India, we have painstakingly created a proprietary “dataset” for "Rishi Pranali," a novel symbolic language. The deliberate goal of this large-scale project is to reduce the burden and improve productivity of the Indian legal system. The fundamental aspect of this project is the development of distinctive fonts made especially for Rishi Pranali, guaranteeing the precise portrayal and smooth incorporation of its symbols into our framework. In order to improve our dataset's readability and adaptability, we have carefully added components from an already-existing Hindi dataset that we obtained from Kaggle, a well-known source of a wide range of datasets. This extensive data set acts as a link between modern data-driven technologies and traditional symbolic languages, as well as a fundamental component for the identification and comprehension of Rishi Pranali. Through the integration of established data science principles with the subtleties of native languages, our initiative aims to provide the Indian legal system with a state-of-the-art instrument. With its sophisticated answer to the intricacies present in the Indian legal system, this marriage of language and technology has the potential to completely transform legal documentation. Our mission is to make the legal system more user-friendly and effective while also advancing the larger objective of enhancing the effectiveness and efficiency of the Indian judiciary through creative, data-driven methods.

Chapter 3

System Analysis & Design

3.1 Requirement Specifications

3.1.1 Dataset Used

The "Hindi Character Recognition" Kaggle dataset is an extensive collection created to further progress handwriting analysis and optical character recognition (OCR) for the Hindi language. This dataset, which contains over 70,000 handwritten character images, offers scholars, developers, and machine learning enthusiasts a rich and varied collection. The inclusivity of the dataset, which captures a wide range of handwriting styles and variations indicative of the inherent diversity in how Hindi characters are written, is one of its distinguishing features. All the images in the dataset have been carefully labeled with the appropriate character, providing a structured basis for supervised learning techniques.

The dataset is more useful due to its careful organization, which makes it easier to train and test machine learning models that are intended to produce accurate Hindi character recognition.

One of the dataset's greatest strengths is its diversity, which includes characters written by authors with varying writing styles and in a variety of contexts. This diversity gives machine learning models flexibility, allowing them to accurately identify and categorize Hindi characters in situations where handwriting styles may differ greatly. This dataset can be used by scholars and professionals working in the fields of computer vision, image processing, and machine learning to further the development of Hindi character recognition systems.

The Kaggle "Hindi Character Recognition" dataset is an essential tool for improving OCR technology, improving language understanding models, or furthering the field of handwriting analysis. Its abundance of labeled data, diversity, and organization make it an invaluable resource for advancing innovation and advancement in the field of Hindi character recognition, which is in line with Kaggle's overarching goal of encouraging cooperative and significant data-driven research.

3.1.2 Python

Python is an interpreted, high-level, general-purpose programming language. Python is simple and easy to read syntax emphasizes readability and therefore reduces system maintenance costs. Python supports modules and packages, which promote system layout and code reuse. It saves space but it takes slightly higher time when its code is compiled. Indentation needs to be taken care of while coding. Python does the following:

1. Python can be used on a server to create web applications.
2. It can connect to database systems. It can also read and modify files.
3. It can be used to handle big data and perform complex mathematics.
4. It can be used for production-ready software development.

Python has many inbuilt library functions that can be used easily for working with machine learning algorithms. All the necessary python libraries must be pre- installed using "pip" command.

3.1.3 Python Libraries

- **Anaconda:** Anaconda is a comprehensive, open-source distribution of Python and R programming languages that is designed for data science, machine learning, and scientific computing. It serves as a versatile platform encompassing a wide array of tools and libraries, making it a go-to solution for developers, data scientists, and researchers.

- **EasyOCR:**

  EasyOCR is an intuitive optical character recognition (OCR) tool made to make text extraction from images simple. EasyOCR is a tool that helps users turn scanned documents, images, or screenshots into editable and searchable text. It has an intuitive interface and strong functionality. Its multilingual support makes it adaptable to a wide range of uses, including text recognition, data extraction, and document digitization. EasyOCR's accuracy and user-friendliness make it a viable option for both individuals and companies looking for effective text extraction from visual content.

- **Matplotlib**

  A flexible Python package called Matplotlib can be used to create interactive, animated, and static visualizations in a number of different formats. With the help of its extensive plotting options, users can create excellent graphs, charts, and plots. For data visualization, Matplotlib is widely used in data science, scientific research, and other domains.

- **Cv2**

  A well-known Python computer vision library is called OpenCV (Cv2). It gives developers the means to work with visual data by offering tools and functions for processing images

and videos. Cv2 is an important tool for computer vision applications because it makes tasks like object detection, facial recognition, and image manipulation simpler.

- **Pylab**

  For convenience in data visualization and analysis, Pylab is a Python module that brings together the features of NumPy, Matplotlib, and other scientific libraries into a single namespace. It is a useful tool for scientific and engineering applications since it offers an easy-to-use interface for interactive plotting and numerical computation.

- **Ipython**

  The Python programming experience is improved by IPython, an interactive command-line shell. It has a strong and intuitive user interface that supports functions like syntax highlighting, code autocompletion, and quick access to documentation. Python is a useful tool for developers and data scientists because it makes interactive coding, data visualization, and exploration easier.

3.1.4 PyCharm

For Python programming, PyCharm is a well-liked integrated development environment (IDE). It provides sophisticated code analysis, debugging, and project navigation tools, and it was developed by JetBrains. Databases, web development frameworks, and popular version control systems are all supported by PyCharm. Its intuitive interface and clever coding support increase the efficiency of Python development.

3.1.5 Jupyter Notebook

Users can create and share documents with live code, equations, visualizations, and narrative text using the open-source web application Jupyter Notebook. Because it supports multiple programming languages and encourages interactive and exploratory computing, data science, research, and education have come to rely on it heavily.

3.1.6 Illustrator

Adobe Inc. created Adobe Illustrator, a vector graphics editor. For the creation of illustrations, logos, icons, and other visual content, it is widely used. Illustrator, which is well-known for its

accuracy and adaptability, enables designers to work with scalable graphics and produces high-quality results for both print and digital media.

3.1.7 Photoshop

Adobe Photoshop is a potent graphics editing program that's frequently used for design and image manipulation. It provides a selection of tools for digital painting, photo retouching, and graphic design. Because of its versatility and advanced features for accurate and creative editing, Photoshop is a go-to program for both professionals and enthusiasts.

3.1.8 Caliber

The term "caliber" describes the diameter of a bullet or the internal diameter of a gun barrel. It is frequently used to determine the size of firearms and the compatibility of ammunition. The phrase is essential to firearm specifications because it influences overall weapon performance and aids in selecting the right ammunition for a given gun.

3.2 System Design

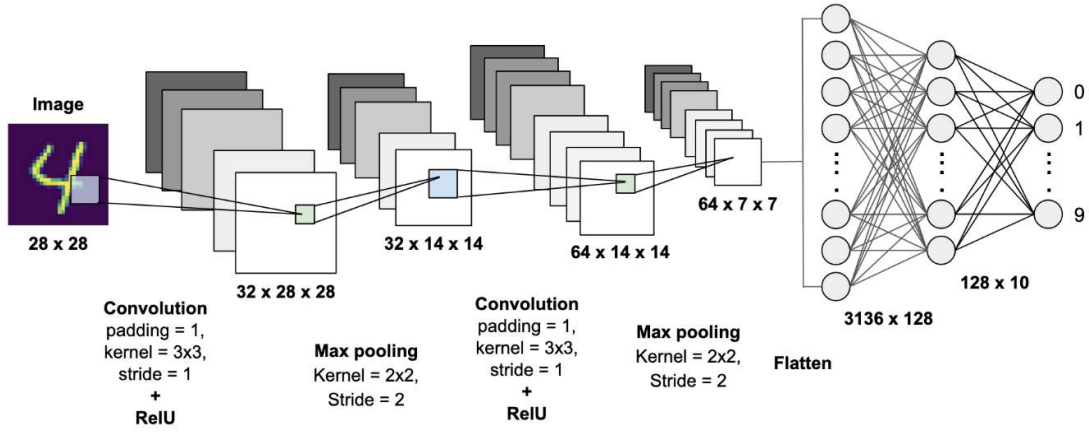

Figure 3.1: System Design

3.2.1 System Architecture

The term "system architecture" describes the high-level organization of a complex system, outlining its individual modules or components and how they work together to achieve

particular goals. It includes the rules, patterns, and design principles that specify how different components of a system interact with one another. An effective blueprint for developing, implementing, and maintaining the system is provided by a clearly defined system architecture. It entails making choices regarding hardware and software components, communication protocols, data flow, and system organization as a whole. Development is based on the architecture, which guarantees maintainability, scalability, and reliability. By specifying a system's behavior and structure, developers can work together more easily and systematically to create hardware or software solutions that are reliable and flexible.

3.2.2 OCR Architecture

The architecture of optical character recognition (OCR) consists of a methodical configuration of parts intended to transform handwritten or printed text into data that can be read by machines. There are usually multiple important phases in the process. The text is first captured by image acquisition, and then the image quality is improved by pre-processing procedures. In order to prepare the text for feature extraction, the segmentation step separates the text into individual characters or words. By locating distinguishing elements such as corners or edges, feature extraction creates a set of traits unique to each character. In the recognition stage, these features are interpreted and characters are identified using machine learning algorithms, like neural networks. Post-processing improves accuracy and refines the results by fixing mistakes. The complexity of OCR architectures varies; from conventional rule-based systems to sophisticated deep learning models, each designed for particular applications like document scanning, text extraction, or handwriting recognition.

3.3 Methodology/Process

The elaborate process of creating the Rishi Pranali font began with a manual design stage in Illustrator that involved a pen and notepad. This first stage made it possible for the symbolic language's visual identity to develop naturally, using hand strokes to convey subtleties. A second step was to carefully clean up the designed fonts in Illustrator itself to guarantee perfect accuracy. The goal of this refining process was to improve visual consistency and clarity, which would serve as the cornerstone for a polished and unique font.

A crucial step in the font creation process was the integration of Unicode. Key mapping was used to assign and allocate each symbol in the Rishi Pranali font through this uniform encoding system. This methodical approach adds to the fonts' compatibility and versatility by guaranteeing that they can be used and represented accurately across a range of platforms.

After the painstaking design, cleanup, and Unicode assignment, the application of a particular software “caliber” is the next step. The conversion of the designed fonts into Windows installable formats is made possible in large part by this software. Users can enjoy a user-friendly and accessible experience with the Rishi Pranali font thanks to the conversion process, which guarantees the font's seamless integration into Windows operating systems.
To put it simply, the Rishi Pranali font is the result of a multifaceted process that balances technical encoding, digital refinement, and hand design. This thorough approach guarantees the symbolic language's usefulness and functionality in the digital sphere, especially in the Windows environment, while also capturing its artistic essence. The end product is a precisely designed and technologically advanced typeface that perfectly captures Rishi Pranali's distinct personality.

3.3.1 Data creation

For our project, the dataset was created using a methodical and careful process to guarantee the validity of our model's training and assessment. To start, we created a set of 28 characters, each of which stood for a different concept in the symbolic language. We generated a great deal of data for each of these characters, which led to the creation of 12 different sets of images that were carefully sorted into the "train" folder. This large-scale variation was intended to provide the model with a variety of examples of each character to help it gain a complete understanding of their visual cues during the training phase.
Concurrently, we selected six more sets of pictures for every character, saving them in the "test" folder. This meticulous data partitioning guarantees a trustworthy assessment of the model's performance on hypothetical cases, gauging its capacity for generalization. An extensive evaluation of the model's efficacy and accuracy is made possible by the purposeful selection of six variants for each character in the testing dataset. We carefully cropped every image in our dataset to improve its uniformity and quality. This phase was essential for getting rid of extraneous details and concentrating on each character's main representation. Furthermore, we used Photoshop to perform a thorough cleanup, enhancing the images' visual details and maximizing their suitability for testing and training. The laborious process of creating the data reflects our dedication to giving the model a varied and well-structured set of examples for strong learning. The training dataset's intentional variation guarantees that the model can generalize to various visual representations of each character. Concurrently, the testing dataset, comprising of well-chosen variations, functions as a meticulous standard for assessing the model's efficacy in practical scenarios. This thorough and meticulous approach

to data creation provides a strong basis for our character recognition model's ensuing training and testing stages.

3.3.2 Keras 5 Layer OCR Model

The Architecture can be broken down into 5 layers:

1. Input Layer

```
model.add(Convolution2D(32, (3, 3), input_shape=(image_height,
image_width, channels), activation='relu'))
```

- This is a convolutional layer (Conv2D) with 32 filters.
- The (3, 3) represents the size of the convolutional kernel.
- input_shape is the shape of the input data. In this case, it assumes an image with dimensions image_height x image_width and channels color channels.
- The activation function used is Rectified Linear Unit (ReLU), which introduces non-linearity to the model.

2. Max Pooling Layer

```
model.add(MaxPooling2D(pool_size=(2, 2)))
```

- This is a max pooling layer (MaxPooling2D) with a pool size of (2, 2).
- Pooling is a down sampling operation that reduces the spatial dimensions of the input data.

3. Convolutional Layer

```
model.add(Conv2D(64, (3, 3), activation='relu'))
```

- Another convolutional layer with 64 filters and a (3, 3) kernel.
- It follows the ReLU activation function.

4. Max Pooling Layer

```
model.add(MaxPooling2D(pool_size=(2, 2)))
```

- Another max pooling layer with a pool size of (2, 2).

5. Flatten Layer

```
model.add(Flatten())
```

- The flatten layer converts the 2D feature maps into a 1D vector. It's a transition from convolutional/pooling layers to fully connected layers.

6. Fully Connected Layer

```
model.add(Dense(128, activation='relu'))
```

- A fully connected layer (Dense) with 128 neurons.
- Activation function is ReLU.

7. Output Layer

```
model.add(Dense(num_classes, activation='softmax'))
```

- Another fully connected layer, but with the number of neurons equal to the number of output classes in your OCR task.
- The activation function is softmax, which is often used in multi-class classification problems to produce probability distributions over the classes.

Chapter 4

Feature and Extraction

- The creation of the Rishi Pranali dataset showcases a number of unique features and extraction techniques that add to its comprehensive and inventive nature.

Features:

1. **Cultural Integration:** The project fosters a harmonious integration of technological innovation and cultural heritage by blending traditional symbolic languages, represented by Rishi Pranali, with modern data-driven methodologies.
2. **Bespoke Font Design:** The development of custom typefaces for Rishi Pranali guarantees precise encoding and identification of its distinct symbols, augmenting the genuineness and efficiency of the dataset.
3. **Hindi Dataset Integration:** Adding components from an already-existing Hindi dataset that was sourced from Kaggle enhances the dataset's readability and adaptability, making it applicable outside of Rishi Pranali.

Extraction Methodologies:

1. **Collaborative Approach:** Taking a cue from Wikipedia highlights the importance of creating datasets collaboratively, which is consistent with the open-source philosophy. An initiative driven by the community to investigate and comprehend symbolic languages is fostered by the collaborative environment.
2. **Bespoke Font Integration:** A painstaking process of design and implementation goes into integrating custom fonts into the system, guaranteeing that the symbols of Rishi Pranali are faithfully portrayed and smoothly incorporated into the dataset.
3. **Kaggle Dataset Amalgamation:** A pre-existing Hindi dataset on Kaggle is strategically combined with elements that have been carefully chosen and modified, adding to the dataset's diversity and richness.

- The features and extraction techniques of the project demonstrate a comprehensive and cooperative approach, making use of current resources while incorporating novel elements to produce a dataset that is not only technically sound but also culturally grounded and adaptable.

Chapter 5

Result and Discussion

The completion of our project to create an extensive dataset for "Rishi Pranali," a symbolic language designed for the Indian legal system, represents a noteworthy accomplishment at the nexus of technological innovation and linguistic preservation. The initiative's outcomes

demonstrate not only the technical facets of dataset generation but also the wider consequences for cultural heritage preservation, the effectiveness of the legal system, and the progress of linguistic research. Our project's successful integration of custom fonts created just for Rishi Pranali is one of its main outcomes. This crucial stage guarantees that the language's distinctive symbols are accurately represented, encapsulating its spirit and complexities. By designing unique fonts, we have paved the way for the development of a dataset that faithfully captures the linguistic subtleties of Rishi Pranali, helping to maintain this symbolic language in a digital environment.

The addition of components from a Hindi dataset that we downloaded from Kaggle enhances our dataset's readability and adaptability. This deliberate combination provides a wider range, allowing for variances and guaranteeing the dataset's suitability in a variety of situations. The outcomes highlight the flexibility of our methodology and indicate that it can be applied in scenarios outside of Rishi Pranali. Technically speaking, the Rishi Pranali dataset is a useful tool for the Indian legal system because of its successful creation. This dataset's representation of symbols and meanings provides a standardized and effective transcription method that has the potential to completely transform legal documentation procedures.

The project's outcomes highlight the usefulness of linguistic studies in a legal setting and are consistent with the overarching objectives of legal frameworks' digital transformation. Additionally, the project's outcomes advance the field of linguistic studies. For scholars and linguists, the dataset acts as a repository, enabling in-depth investigation and interpretation of the distinctive meanings and symbols present in Rishi Pranali. This advances our knowledge of symbolic languages academically and creates new opportunities for study and advancement in the area.

Ultimately, our project's outcomes go beyond just producing a dataset—rather, they embody a fusion of technology and culture. The incorporation of Rishi Pranali into a digital framework is a step toward innovation, language heritage preservation, and process efficiency improvements in the legal domain. The accomplishment of this project demonstrates the transformative power

of language preservation within the rapidly changing landscape of digital advancements, and it resonates with the potential for technological solutions to bridge cultural divides.

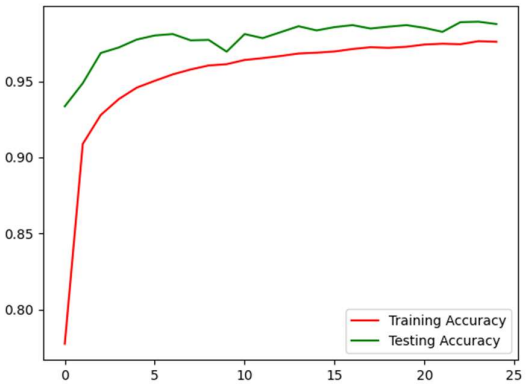

Figure 5.1: Success Rate

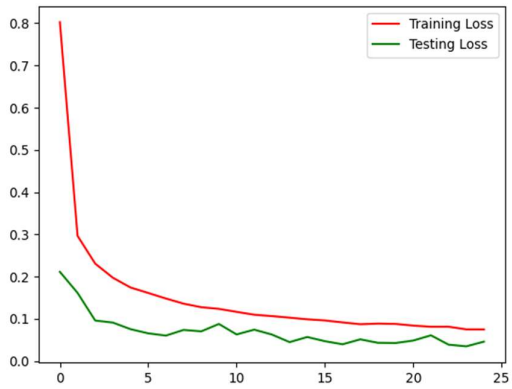

Figure 5.2: Lose Rate

F
i

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 conv2d (Conv2D)             (None, 30, 30, 32)        320

 batch_normalization (Batch  (None, 30, 30, 32)        128
 Normalization)

 max_pooling2d (MaxPooling2  (None, 15, 15, 32)        0
 D)

 conv2d_1 (Conv2D)           (None, 13, 13, 32)        9248

 batch_normalization_1 (Bat  (None, 13, 13, 32)        128
 chNormalization)

 max_pooling2d_1 (MaxPoolin  (None, 7, 7, 32)          0
 g2D)

 conv2d_2 (Conv2D)           (None, 5, 5, 64)          18496

 batch_normalization_2 (Bat  (None, 5, 5, 64)          256
 chNormalization)

 max_pooling2d_2 (MaxPoolin  (None, 3, 3, 64)          0
 g2D)

 conv2d_3 (Conv2D)           (None, 1, 1, 64)          36928

 batch_normalization_3 (Bat  (None, 1, 1, 64)          256
 chNormalization)

 max_pooling2d_3 (MaxPoolin  (None, 1, 1, 64)          0
 g2D)

 flatten (Flatten)           (None, 64)                0

 dense (Dense)               (None, 128)               8320

 batch_normalization_4 (Bat  (None, 128)               512
 chNormalization)

 dense_1 (Dense)             (None, 64)                8256

 batch_normalization_5 (Bat  (None, 64)                256
 chNormalization)

 dense_2 (Dense)             (None, 46)                2990

=================================================================
Total params: 86094 (336.30 KB)
Trainable params: 85326 (333.30 KB)
Non-trainable params: 768 (3.00 KB)
_________________________________________________________________
None
```

Figure 5.3: Output

Chapter 6

Conclusion

Finally, the goal of our ground-breaking project has been to develop a transformative dataset for "Rishi Pranali," a symbolic language that has been carefully customized to enhance workflows in the Indian legal system. Our initiative was motivated by the abundance of information about linguistic nuances, cultural context, and the potential applications of symbolic languages. It was founded on collaborative insights derived from Wikipedia.

The creation of unique fonts specifically made for Rishi Pranali was the central process of our dataset construction, guaranteeing accurate representation and smooth system integration. To enhance our work, we carefully included components from a Hindi dataset that we obtained from Kaggle, a reliable source of a wide variety of datasets. Our dataset is positioned as a link between ancient linguistic traditions and cutting-edge technology thanks to the combination of traditional symbolic languages with modern data-driven methodologies. This project is significant not only because it has the potential to transform legal documentation in the Indian judiciary but also because it will help preserve and comprehend distinct linguistic expressions in a larger context. The initiative is motivated by a dedication to creativity, information exchange, and flexibility in the ever-changing field of digital innovations. In addition, the development of this dataset represents a forward-thinking step toward improving legal processes' accessibility and effectiveness. Through the smooth integration of contemporary data science concepts with the subtleties of indigenous languages, our project seeks to equip legal practitioners with a tool that can adapt to the changing needs of the digital age. Our project, in its essence, is a combination of technology and tradition, embodying the spirit of advancement while honoring cultural legacy. Beyond its technical aspects, the development of the Rishi Pranali dataset is a step toward promoting cultural inclusivity, creativity, and efficiency in legal proceedings. It also has significant implications for the future of language preservation and technological integration in the legal field. Our project adds to the greater discussion on the incorporation of various linguistic traditions into modern digital workflows as we traverse the interface of language, technology, and legal frameworks. Through the promotion of a more profound comprehension of symbolic languages, our goal is to enable a flexible and inclusive method of legal documentation, thus creating the foundation for future developments at the nexus of language and technology in the legal domai

Chapter 7

References

1. "Web Browser",
   Web browser - Wikipedia

2. "Open AI",
   https://chat.openai.com

3. "Kaggle",
   https://www.kaggle.com/

4. "Books",
   Hindi Sanket Leepi: https://archive.org/details/in.ernet.dli.2015.538685

5. "Images".
   https://www.pictures.microsoft.com

Chapter 8

Contributors

1. **Sapeksh Pareek**
   Enrolment Number: 211B274
   Email: 211B274@juetguna.in
   Personal Email: sapekshpareek136@gmail.com
   Address: Khilchipur, M.P

2. **Sarthak Nagar**
   Enrolment Number: 211B276
   Email: 211B276@juetguna.in
   Personal Email: sarthaknagarjii@gmail.com
   Address: Khilchipur, M.P

3. **Shivansh Bhatnagar**
   Enrolment Number: 211B296.
   Email: 211B296@juetguna.in
   Personal Email: bhatnagarshivansh19@gmail.com
   Address: Gwalior, M.P