Chapter 5

System Implementation

This chapter examines the overall plan of this project. The chapter too gives a portrayal of the architecture of the framework conjointly the detailed plan issues of the components of the system.

5.1 Proposed system

We will do this by including new types of information into models and by considering how to compose simple components into highly active systems. Mainly we will focus on three areas of scene text recognition, each with a decreasing number of prior conjectures. Firstly, we will introduce two techniques for character recognition, where word and character bounding boxes will be used. Next we will look at word recognition, where only word bounding boxes will be used. We want to develop a new technique for fragmenting text for these images called bilateral regression segmentation. Lastly, we will remove the assumption that words have been located and describe an end-to-end system that detects and recognizes text in any natural scene image.

5.1.1 Segmentation

Documents horizontal projection profile is found to separate the lines in the image. Histogram of the number of ON pixels along every row of the image is the horizontal projection profile. The letters in Kannada are composed by attaching to the glyph of a consonant, the glyphs of the vowel modifiers and the glyphs of the consonant conjuncts. If we considered all the combination, then building the classifier of these numbers of character is very difficult. So our strategy is that we will segment the word into its constituents, i.e. the base consonant, the vowel modifier and the consonant conjunct. It's very difficult to achieve this. If we have good look on the Kannada word, we will see that for extracting glyph of a consonant the glyphs of the vowel modifiers and the glyphs of the consonant conjuncts we can divide the character into two zones. Top zone: Top zone mainly consist of main portion of the character. It includes base consonant or vowels or some vowel modifiers. Bottom zone: Bottom zone consists of glyphs for the consonant conjuncts. Here by using connected component method, we are first counting the number of consonants, vowels, vathu or vowel modifiers present in the text line. Connected component method is an algorithmic application of graph theory, where subsets of connected components are uniquely labeled based on a given heuristic. Connected component labeling is used in computer vision to detect connected regions in binary digital images, although color images and data with higher-dimensionality can also be processed. In connected component method connectivity checks are carried out by checking the labels of pixels that are North-East, North, North-West and West of the current pixel. Then we extracting that characters separately and send it for character recognition.

5.1.2 Canny Edge Detector

Edge detection is image processing technique for finding the boundaries of object within image. Edge detection is used for image segmentation and data extraction. The main criteria of canny edge detector is

i. Low error rate -good detection of only existent edge.
ii. Good localization -distance between edge pixel detected and real edge pixel need to be minimized.
iii. Minimal response –only one detector response per edge.

It includes five steps,

i. Filter image with derivative of Gaussian
ii. Find magnitude and orientation of image
iii. Taking magnitude and performing some local maxima i.e., non maximum suppression .here multiple pixel are thinned into single pixel
iv. Linking and thresholding is performed on Gradient image. Here grouping of pixel is done based on low threshold and high threshold. High threshold is used to start curve and low threshold is used to continue curve
v. Finally edge map is obtained by considering only the edges that are connected to strong edges.

5.1.3 Training a SVM Model

SVM Model means Support Vector Machine; this model is used for training a dataset using supervised model for classifying and regression of data through learning algorithm that analyzes the data. This model is also used for mapping points in the space and also for linear and non-linear classification. As this model uses supervised learning in which a particular input is given for the required output which organizes different algorithm and training of data. In supervised learning we have to analyze different step as follows:

i. First is determining the different training dataset and analyzing the dataset whether it is handwritten data or typed.
ii. Collecting of different dataset which includes set of input objects and output according to it.
iii. Analyzing the feature representation of particular input and also the features on which input dataset depends like dimension, no of vectors and feature of vector.
iv. Analyzing the structure of the function in model and algorithm like instead of SVM we can choose decision trees.
v. Running the algorithm for the featured function and completing the design with the certain parameters.
vi. Last we have to calculate the accuracy for the given dataset as here we are giving the positive and negative dataset.

Training a model can be done once for the dataset and we can proceed for the testing of different dataset.

As here we are doing the training for the 251 positive and negative dataset. During training the dataset is mapped and particular accuracy is obtained.

5.1.4 Testing a SVM Model

After testing a SVM model then training part comes, where a particular dataset is mapped and checked with a regular dataset whether it is mapping or not. In this method we use bounding box technique for mapping of the data. In this data checked according to the no of lines and mapped for different lines in separate windows. If the data is mapped it will show the boundary for particular consonants and vowels and data segmented for particular line. If data doesn’t match the data will be blurred or unbounded, it happens in the case when the input sample is not clean or blurry image is there. So, clarity of the image is one of the important factors while training.

5.2 System Architecture

System architecture is a conceptual demonstration that characterizes the structure, behavior, and more views of a framework. A design portrayal could be a formal portrayal and representation of a framework, organized in a way that underpins thinking almost the structures and behaviors of the framework. A framework engineering can contain framework components that will work together to execute the generally system.

The following Figure shows the architecture of the system.

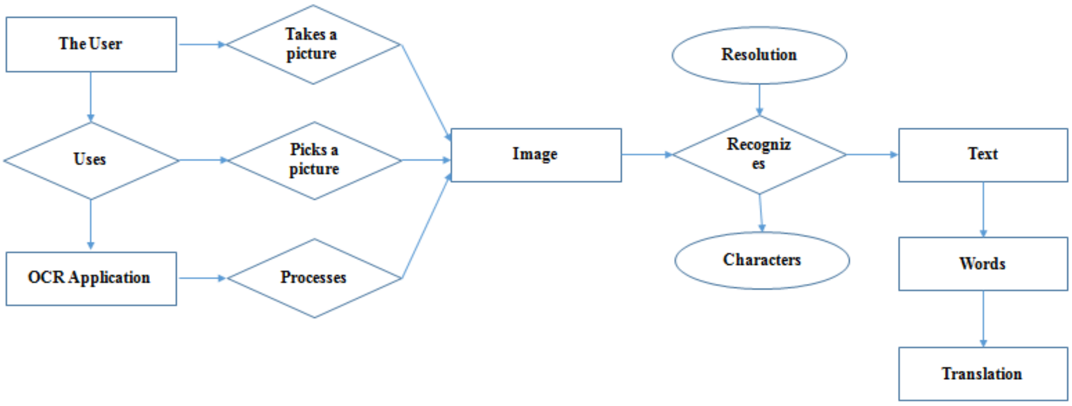

Figure 5.1: System Architecture

Our system architecture consists of following components.

- End user
- Graphical User Interface(GUI)
- Operating System

Inside the domain of the venture, we have four modules which are intuitively to the client to be specific Segmentation, Training, Testing and OCR. When the client interacts with one of these modules a yield is delivered. The division module is advance separated into six parts, which in turns gives particular yield.

The above figure shows the system architecture of this project. It contains all the essential module and tools which are required to develop the project. It gives us the blue print of the projects, thereby showing the flow of work.

5.2.1 Flow Diagram

This is the flow diagram of training and testing of standard dataset and to that of sample images taken from nature. It shows how each phase will work at different stages.

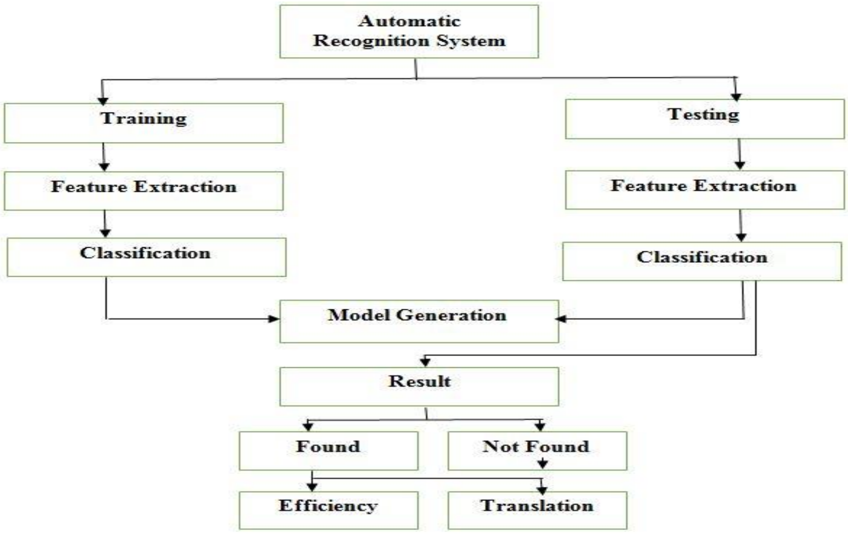

Figure 5.2: Flow Diagram

The above figure shows us the flow diagram of the project for training and testing part. The training part undergoes ‘feature extraction’ and then ‘classification’ which in turn goes to model generator. The testing part also undergoes the same process as training. The extension of testing after classification is result. After this phase the sample image is compared to standard annotated database. It gives the identified characters in notepad file.

5.2.2 Sequential Flow Diagram

This phase shows the linear flow of the project. Firstly, the input image is browsed and sent for further processing which is listed below.

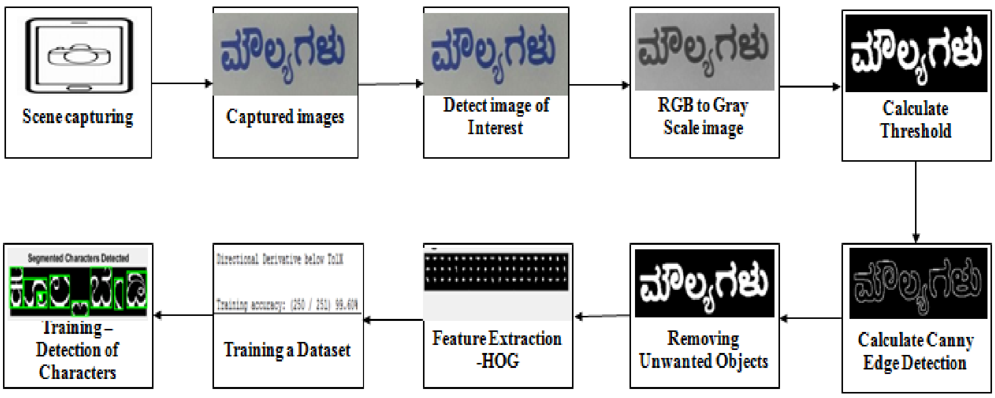

Figure 5.3: Sequential Flow Diagram

The linear steps are as follows:-

i. Scene Capturing
ii. Captured Image
iii. Detect Image of Interest
iv. RGB to GrayScale Image
v. Calculate Threshold
vi. Calculate Canny Edge Detection
vii. Removing Unwanted Objects
viii. Feature Extraction- HOG
ix. Training a Dataset
x. Training- Detection of Characters

5.2.3 End User

An end user is an individual who eventually employments or is planning to eventually utilize an item. The conclusion client stands in differentiate to clients who bolster or keep up the item such as sysops, framework directors, database directors, data innovation specialists, computer program professionals and computer specialists. Conclusion clients ordinarily don't have the technical understanding or aptitude of the item originators, a reality that's simple for originators to disregard or neglect, driving to highlights with which the client is disappointed.

5.2.4 Operating System

Operating System is low-level program that underpins a computer's essential capacities, such as planning errands and controlling peripherals. It acts as halfway between end user and the system created and introduced within the O.S.

5.2.5 Data Set of Natural Image

We have captured 200 images from mobile phone and digital camera. Sample images are depicted in Figure2.2 We have applied two methods of segmentations: canny edge detection technique and rectangular bounding boxes as shown in Figure 6. Both techniques are equally likely to be good. The canny edge detection technique detects objects other than characters and digits.

Figure 5.4: Canny Edge and Rectangular Bounding Boxes

Out of 20 images of natural scene in our database, 209 numbers of characters are present. The proposed system has identified 185 numbers of characters. The success of identified characters is tabulated separately with percentage ratio in Table 5.1.

Image ID	No. Of Characters	No. Of Characters Identified	Performance
1	6	3	50%
2	2	1	50%
3	4	4	100%
4	8	8	100%
5	7	7	100%
6	2	1	50%
7	49	49	100%
8	49	49	100%
9	3	0	0%
10	10	10	100%
11	14	14	100%
12	9	9	100%
13	5	5	100%
14	6	6	100%
15	4	4	100%
16	4	1	25%
17	10	10	100%
18	4	4	100%
19	7	0	0%
20	6	0	0%
TOTAL	209	185	88.52%

Table 5.1: Performance Chart

5.3 Features

We have tried features like Shape Context and Geometric Blur, these are based on shape and edge detection. SVM classifier technique has also been used for its higher efficiency and faster processing unlike SIFT which is comparatively slow. The methods like canny edge detection, change of RGB image to Gray scale image, calculation of threshold for each image, highlights utilized for speaking to surface, such as channel reactions, patches and Spin Images have moreover been utilized [20]. We have also tried to remove the unwanted objects like graphics from foreground and background from the natural scene image. We have analyzed various commonly used parameters and feature detection technique used for each descriptor which have been described below. We have also tried to remove the unwanted objects like graphics from foreground and background from the natural scene image. We have analyzed different commonly utilized parameters and highlight location procedure utilized for each descriptor which has been depicted underneath.

5.3.1 Convert RGB to Gray Scale [22] To convert RGB to grayscale, the average of all the three i.e. R, G and B is computed. To do so we add R with G with B and then divide it by 3 to obtain the grayscale. To calculate the rgb scale from gray scale image we use equation (1).

$$Gray(i,j)=0.29*rgd(:,:,1)+0.59*rgb(:,:,2)+0.11*rgb(:,:,3) \quad ..... (1)$$

5.3.2 Geometric Blur (GB) [2] Geometric blur is simply an average over geometric transformations of a signal. It is done by sampling method called feature extraction which is same as SC. It is isolated into distinctive locales and at that point the edge introductions are checked with distinctive obscure figure

5.3.3 Calculating Threshold [28] Calculation of threshold of an image is done by separating a picture into closer view and foundation independently. This handle changes over the grayscale picture into binary picture. Equation (2) computes threshold value.

Level=graythresh(Image)

BW=im2bw(Image, level); ……(2)

5.3.4 Shape Contexts (SC) [1] SC is a feature descriptor. It is used for object recognition and description of shape that permits measuring shape similarity. We do it by Sobel edge detection technique using log-polar histogram. We use histogram of oriented gradients (HOG) technique as well.

Function F=getHistogramFeatures(Points,Points1,Normals,Option) …..(5)

5.3.5 Canny Edge Detection [24], [23] By the help of Canny Edge Detection algorithm, we detect edges of objects in an image. It is a multi-stage algorithm. It is useful as it extracts structural information.

The algorithm runs in 5 separate steps:

i. Smoothing: Obscuring of the picture to evacuate clamour.
ii. Finding gradients: The edges ought to be stamped where the slopes of the picture has expansive extents.
iii. Non-maximum suppression: Only local maxima should be marked as edges.
iv. Double thresholding: Potential edges are decided by thresholding.
v. Edge tracking by hysteresis: Last edges are decided by stifling all edges that are not associated to a really certain (solid) edge.

5.3.5.1 Smoothing

It is inescapable that all pictures taken from a camera will contain a few sum of commotion. To prevent that commotion is mixed up for edges, clamor must be diminished. In this manner the picture is first smoothed by applying a Gaussian channel. The bit of a Gaussian channel with a standard deviation of = 1.4 is appeared in Equation (1).

$$B = \frac{1}{159} \cdot \begin{bmatrix} 2 & 4 & 5 & 4 & 2 \\ 4 & 9 & 12 & 9 & 4 \\ 5 & 12 & 15 & 12 & 5 \\ 4 & 9 & 12 & 9 & 4 \\ 2 & 4 & 5 & 4 & 2 \end{bmatrix} \tag{1}$$

5.3.5.2 Finding Gradients

The Canny calculation essentially finds edges where the grayscale escalated of the picture changes the most. These ranges are found by deciding slopes of the picture. Slopes at each pixel in the smoothed picture are decided by applying what is known as the Sobel-operator. First step is to surmise the angle within the x- and y-direction individually by applying the kernels shown in Equation (2).

$$K_{\mathrm{GX}} = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}$$

$$K_{\mathrm{GY}} = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} \quad (2)$$

The angle extents (too known as the edge qualities) can at that point be decided as a Euclidean remove degree by applying the law of Pythagoras as appeared in Condition (3). It is in some cases disentangled by applying Manhattan separate degree as appeared in Equation (4) to reduce the computational complexity. The Euclidean separate degree has been connected to the test picture.

$$|G| = \sqrt{Gx^2 + Gy^2} \quad (3)$$

$$|G| = |Gx| + |Gy| \quad (4)$$

Where: Gx and Gy are the angles within the x- and y-directions respectively. It is clear from Figure 3, that a picture of the slope sizes frequently demonstrate the edges quite clearly. However, the edges are regularly wide and thus don't show precisely where the edges are. To create it conceivable to decide this (see Section 2.3), the heading of the edges must be decided and put away as appeared in Equation (5).

$$\Theta = \arctan(|Gy| / |Gx|) \quad (5)$$

5.3.5.3 Non-maximum suppression

The reason of this step is to change over the “blurred” edges within the picture of the slope magnitude to “sharp” edges. Typically it is done by protecting all neighborhood maxima within the angle image, and erasing everything else. The calculation is for each pixel within the slope picture:

i. Circular the slope heading θ to closest 45◦, comparing to utilize of an 8-connected neighbourhood.
ii. Compare the edge quality of the current pixel with the edge quality of the pixel in the positive and negative angle course. I.e. in the event that the slope course is north (theta = 90◦), compare with the pixels to the north and south.
iii. In case the edge quality of the current pixel is biggest; protect the esteem of the edge strength. If not, smother (i.e. evacuate) the esteem.

5.3.5.4 Double thresholding

The edge-pixels remaining after the non-maximum concealment step are (still) checked with their strength pixel-by-pixel. Numerous of these will likely be genuine edges within the picture, but a few may be caused by clamor or color varieties for occasion due to unpleasant surfaces. The only way to discern between these would be to utilize a limit, so that as it were edges more grounded that a certain value would be protected. The Canny edge calculation employs twofold thresholding. Edge pixels more grounded than the tall edge are checked as solid; edge pixels weaker than the low threshold are smothered and edge pixels between the two limits are checked as frail.

5.3.5.5 Edge tracking by hysteresis

Solid edges are translated as “certain edges” and can promptly be included within the final edge picture. Powerless edges are included in case and as it were in case they are associated to solid edges. The logic is of course that clamor and other little varieties are improbable to result in a solid edge (with appropriate alteration of the limit levels). Hence solid edges will (nearly) as it were be due to true edges within the unique picture. The frail edges can either be due to genuine edges or noise/color variations. The last-mentioned sort will likely be dispersed freely of edges on the entire image, and hence as it were a little sum will be found adjoining to solid edges. Powerless edges due to genuine edges are much more likely to be associated specifically to solid edges.

5.3.6 Removal of Unwanted Objects [25] The subject of removing unwanted objects from natural scene images without generating any possible distortion has been handled.

5.3.7 Histogram of Oriented Gradients (HOG) [26], [27] It is a highlight descriptor utilized in computer vision and picture handling for the reason of protest location. The procedure tallies events of slope introduction in localized parcels of a picture. Features = extractHOGFeatures (I) It returns extricated Hoard highlights from a truecolor or grayscale input picture, I. The highlights are returned in a 1-by-N vector, where N is the Hoard highlight length. The returned highlights encode neighborhood shape data from locales inside a picture.

5.3.8 Spin Image [11], [6] It is a two dimensional histogram encoding method for conveyance of picture brightness. The two dimensional of the histogram is d, separate from the center point, and i the escalated esteem. We have utilized d=11 and i=5 for concentrated esteem, coming about in 55-dimensional descriptors.

5.3.9 Maximum Response of Filters (MR8) [18] It is a surface descriptor based giving 8D vectors, on a set of 38 channels but as it were 8 responses.

5.3.10 Patch Descriptor (PCH) [19] It is the least complex thick include extraction strategy. For each position, the crude n × n pixel values are vectorized, producing an n^2 descriptor. We utilized 5×5 patches.