CHAPTER 4
OFFLINE HANDWRITTEN GURUMUKHI WORD RECOGNITION SYSTEM BASED ON HOLISTIC APPROACH AND ADABOOST METHODOLOGY

To recognize a word, segmentation of words into characters can be done which is a tedious task. There are several issues in the process of segmentation like overlapping characters, touching characters, detecting the right segmentation point (segmentation ambiguity), cursive handwriting, etc. Due to such issues, the words are unable to be classified properly; thus, it leads to poor accuracy. Hence, to avoid these segmentation issues, a holistic approach to the recognition of offline handwritten Gurumukhi words has been followed in the present study. For the extraction of significant characteristics from the word images, three features such as zoning features, diagonal features, intersection & open-end points features have been considered. Based on the extracted features, the words have been classified using three classifiers like k-Nearest Neighbor (k-NN), Support Vector Machine (SVM), and Random Forest. Then to enhance the recognition performance, ensemble techniques such as Adaptive Boosting (AdaBoost) methodology and majority voting scheme have been applied. The complete chapter is segregated into six sections. Section 4.1 elaborates the concept of a holistic approach to word recognition. Section 4.2 discusses the feature extraction techniques and section 4.3 demonstrates the working of AdaBoost methodology. The system performance has been evaluated based on the experimental results which are presented in section 4.4. The comparison of the present approach with the existing approaches followed by syntactic analysis is mentioned in section 4.5. Finally, section 4.6 summarizes the whole chapter.

4.1 HOLISTIC APPROACH

In the holistic approach, the word itself is considered as an individual entity and the whole word is recognized from its comprehensive shape without segmenting the word into its individual characters. Due to the absence of the segmentation process, this approach is also known as the segmentation-free approach. In the literature, the holistic approach has gained sufficient attention in the field of word recognition as an interesting and more straightforward solution. The motivational factors behind the use of a holistic approach to word recognition are discussed as below:

- The algorithms based on a holistic approach are mathematically efficient.
- In case of poor handwriting, the individual characters cannot be differentiated but the comprehensive shape of the word can be preserved using a holistic approach.
- This approach is supported by psychological studies of human reading which specify that humans utilize characteristics of word shape in reading.
- It is executed better as compared to segmentation-based approach for known, fixed, and small-sized lexicon.

4.2 FEATURE EXTRACTION TECHNIQUES

To recognize offline handwritten Gurumukhi words based on a holistic approach, three features were extracted from the word images such as zoning features, diagonal features and intersection & open-end points features which are discussed in the following sub-sections. Kumar *et al.* (2014b) also utilized these features to recognize offline handwritten Gurumukhi characters, which is the motivating factor behind the use of these features for this Gurumukhi word recognition system.

4.2.1 Zoning features

In this feature extraction technique, the foreground pixels corresponding to $4^{(L)}$ zones are obtained, where L represents the current level of the word image. For the present work, at first one feature ($4^{(0)}$) was considered from the whole word image. Then the word image was partitioned into 4 zones (4^{1}), which were further partitioned into 4 zones, thus leading to total 4×4=16 zones ($4^{(2)}$). This partitioning continued down to 64 zones ($4^{(3)}$) by splitting each of the 16 zones into 4 zones. It resulted into total 1+4+16+64=85 zones. Thus, we extracted total 85 zoning features from the pattern characteristics or pixel's density of the computed zones.

4.2.2 Diagonal features

Corresponding to the zones, diagonals features were extracted which were then averaged to attain the single value of each zone. By dividing the word image into 85 zones as discussed in zoning features, we extracted 85 (1+4+16+64=85) diagonal features for the present work.

4.2.3 Intersection & open-end points features

At first, the word image was divided into 85 zones as discussed in zoning features. After getting 85 zones, the intersection & open-end points features were calculated from these zones. Thus, we extracted a total of 170 intersection & open-end points features from the considered zones that comprised 85 intersection features and 85 open-end points features.

4.3 ADABOOST METHODOLOGY

AdaBoost is an abbreviation of Adaptive Boosting, which was proposed by Freund and Schapire (1997). It is an ensemble algorithm utilized to transform a group of weak learners (classifiers) into a strong learner. The base learners are created with the help of a Decision Tree having one depth and these Decision Trees are called decision stumps. Each sample in the dataset gets assigned some weight that is equal to *1/n*, where *n* refers to the number of training samples. Then based on weighted samples, a weak learner is constructed. As this methodology works for binary classification problems only, outputs are generated by decision stumps as +1.0 or -1.0 value corresponding to the first-class or second-class, respectively. After training, the rate of misclassification (error) is computed as:

$$error = (correct - N)/N \tag{4.1}$$

where correct refers to the number of correctly predicted training samples and *N* refers to the total number of training samples.

The weak learners are appended in a sequence and are trained based on the weighted training samples. This procedure goes on until a predetermined number of weak learners are generated. On arrival of the test sample, every weak learner computes a predicted value as either +1.0 or -1.0, which are then weighted. At last, the addition of the weighted predictions is considered as the prediction of the ensemble model. Based on positive-sum or negative-sum, the prediction is made as first-class or second-class, respectively.

4.3.1 Working of AdaBoost Methodology

As depicted in Figure 4.1, equal weights are assigned to each data point and then decision stump (D1) is applied to classify the data points as + (plus) or - (minus). The vertical line generated by D1 classifies the data points, which predicts three + (plus) as - (minus) incorrectly. So, the higher weights are assigned to these three + (plus) and then given to another decision stump (D2), which try to predict them accurately. The vertical line generated by D2 classifies three + (plus) accurately but misclassifies three - (minus). So, the higher weights are assigned to three - (minus) and then given to another decision stump (D3). Decision stump 3 (D3) predicts these three misclassified observations accurately and the horizontal line generated by D3 classifies + (plus) and - (minus) on the basis of higher weights of the misclassified observations. These three decision stumps are combined to create a strong prediction and thus, the observations are classified quite well in comparison to the individual weak learners.

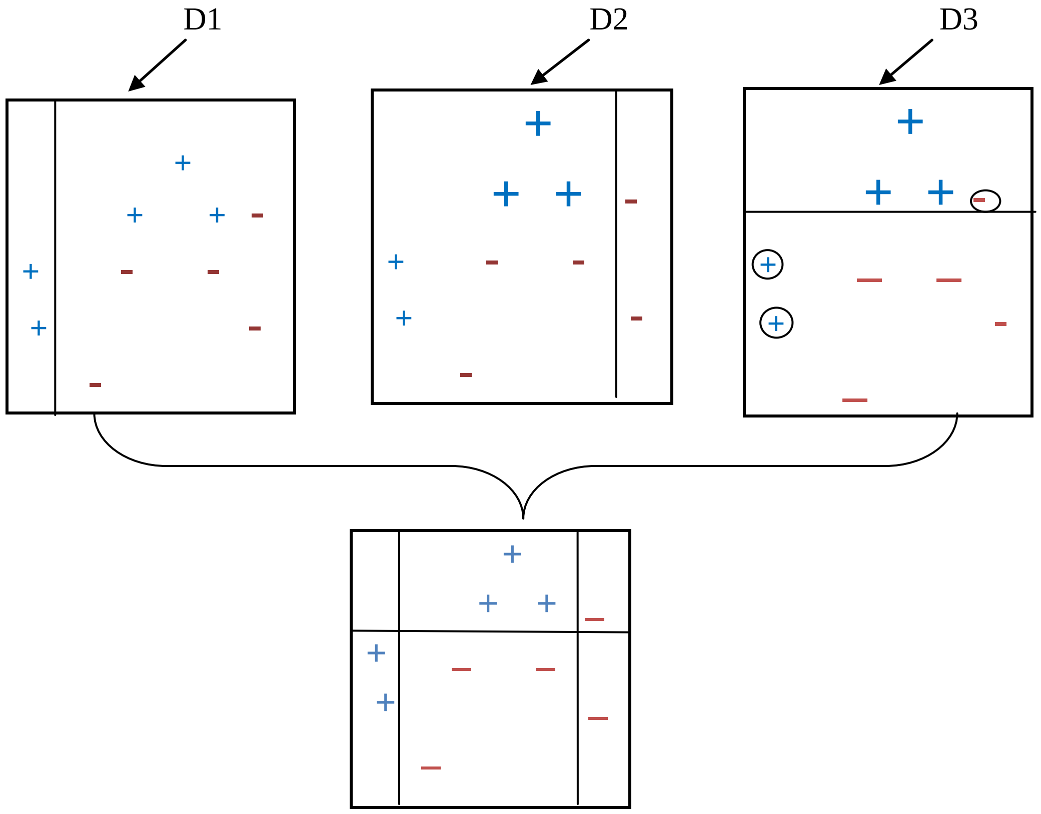

Figure 4.1. Working of AdaBoost Methodology

4.3.2 Pros of AdaBoost Methodology

- It is simple to implement.
- It rectifies the errors of the weak learner iteratively and enhances accuracy based on a combination of weak learners.
- It supports a high level of precision.
- Multiple base classifiers can be used with AdaBoost.
- It is not susceptible to over-fitting.
- It is flexible to be combined with any machine learning technique.

4.3.3 Cons of AdaBoost Methodology

- It is sensitive to noisy data.
- It gets affected by outliers.
- Imbalanced data leads to a decline in classification accuracy.
- It is a time-consuming process due to training.
- It is slower than the XGBoost technique.

4.4 EXPERIMENTAL RESULTS

This section reveals the experimental results based on three features, namely, zoning features (F1), diagonal features (F2), intersection & open-end points features (F3), and three classifiers, namely, k-NN, SVM, and Random Forest. To enhance system performance, AdaBoost methodology and majority voting scheme were employed. Various combinations of features were evaluated to test the performance of classification techniques. The system was evaluated using a dataset of 1,00,000 handwritten samples of Gurumukhi words which correspond to 100 distinct place names of Punjab state. This dataset was gathered from 100 distinct writers belonging to different age groups, places, and professional qualifications. The dataset has been segregated using an 80:20 partitioning strategy where 80% of data (80,000 words) belongs to the training set and the remaining 20% of data (20,000 words) belongs to the testing set. To evaluate the system performance, three parameters such as Accuracy, FAR, and FRR were considered which are discussed below.

- **Accuracy**

Accuracy is interpreted as the division of the number of accurate predictions by the total number of input samples. In binary classification, it is determined in connection with positives and negatives as illustrated below.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \tag{4.2}$$

where,

True Positive (TP): when the true observation is predicted to be true.

False Negative (FN): when the true observation is predicted to be false.

True Negative (TN): when the false observation is predicted to be false.

False Positive (FP): when the false observation is predicted to be true.

- **FAR (False Acceptance Rate)**

FAR is determined as the proportion of identification samples in which unauthorized samples are mistakenly accepted. It is computed as the division of the number of false acceptances by the total number of identification attempts.

$$FAR = \frac{FA}{TA} \tag{4.3}$$

where,
FA = Number of False Acceptances
TA = Total Number of Attempts

- **FRR (False Rejection Rate)**

FRR is determined as the proportion of identification samples in which authorized samples are mistakenly rejected. It is computed as the division of the number of false rejections by the total number of identification attempts.

$$FRR = \frac{FR}{TA} \tag{4.4}$$

where,
FR = Number of False Rejections
TA = Total Number of Attempts

The following sub-sections demonstrate the experimental results based on the considered classifiers.

4.4.1 System performance based on k-NN classifier

k-NN classifier attained maximum accuracy of 75.45% based on a hybrid of zoning, diagonal, and intersection & open-end points based features as depicted in Table 4.1. The system achieved a maximum FAR (0.79%) using zoning features only. Whereas the minimum FAR (0.59%) was attained in three cases such as based on a hybrid of zoning and diagonal features; using a hybrid of zoning and intersection & open-end points based features; and based on a combination of all the three features as demonstrated in Table 4.2. The maximum FRR (30.43%) and minimum FRR (23.96%) were attained based on zoning features only and a hybrid of all the features, respectively, as depicted in Table 4.3. The results based on accuracy, FAR, and FRR are graphically shown in Figures 4.2-4.4.

4.4.2 System performance based on RBF-SVM classifier

RBF kernel-based SVM classifier attained a maximum accuracy of 65.55% based on a combination of all the three considered features as illustrated in Table 4.1. The maximum FAR (1.29%) and FRR (38.41%) were attained based on zoning features only, whereas the minimum FAR (0.99%) and FRR (33.46%) were achieved by considering the hybrid of all the features as illustrated in Table 4.2 and 4.3, respectively.

4.4.3 System performance based on Random Forest classifier

Random Forest classifier attained a maximum accuracy of 76.15% based on a hybrid of all the features as elucidated in Table 4.1. Maximum FAR (0.69%) was attained in four cases such as using only zoning features; only diagonal features; only intersection & open-end points features; and using a hybrid of zoning and diagonal features. Whereas, the minimum FAR (0.59%) was achieved in three cases such as using hybrid of zoning and intersection & open-end points features; hybrid of diagonal and intersection & open-end points features; and using hybrid of all the features as elucidated in Table 4.2. The maximum FRR (26.69%) and minimum FRR

(23.26%) were attained using only zoning features and based on a hybrid of all the features, respectively, as elucidated in Table 4.3.

4.4.4 System performance based on Majority Voting scheme

The majority voting scheme is a hybrid classification scheme that considers the combination of multiple classifiers. A voting scheme is applied for classification when there are multiple models being constructed from the distinct samples of the same training dataset. For the test sample, every model makes some predictions with uniform rights. The prediction that gets the maximum of the votes in comparison to others is considered as the final prediction. This scheme is also termed as plurality voting. For our system, the majority voting scheme attained a maximum accuracy of 83.33% using a hybrid of all the considered features as depicted in Table 4.1. Maximum FAR (0.5%) was attained in four cases such as considering only zoning features; only diagonal features; a hybrid of zoning and diagonal features; and a hybrid of diagonal and intersection & open-end points features. The minimum FAR (0.4%) was achieved in three cases such as employing the only intersection & open-end points features; hybrid of zoning and intersection & open-end points features; and a hybrid of all the features as depicted in Table 4.2. Maximum FRR (19.71%) and minimum FRR (16.27%) were attained based on only zoning features and by amalgamating all the three features, respectively, as depicted in Table 4.3.

4.4.5 System performance based on AdaBoost Methodology

AdaBoost methodology upgraded the system performance by attaining the highest accuracy of 88.78% based on a hybrid of all the three considered features as elucidated in Table 4.1. Maximum FAR (0.5%) was attained using two cases such as based on only zoning features; and only diagonal features. Whilst the minimum FAR (0.4%) was achieved based on five cases such as using only intersection & open-end points features; hybrid of zoning and diagonal features; hybrid of zoning and intersection & open-end points features; hybrid of diagonal and intersection & open-end points features; and hybrid of all the three features as elucidated in Table 4.2. Maximum FRR (15.27%) and minimum FRR (10.82%) were achieved based on only zoning features and using a hybrid of all the three features, respectively, as elucidated in Table 4.3.

Table 4.1. System evaluation based on Accuracy

Features	**Classification Technique**				
	k-NN	**RBF-SVM**	**Random Forest**	**Majority Voting (C1+C2+C3)**	**AdaBoost**
Zoning (F1)	68.78%	60.30%	72.62%	79.79%	84.23%
Diagonal (F2)	71.21%	62.92%	72.82%	81.51%	85.95%
Intersection and Open-end points (F3)	72.62%	64.34%	73.93%	82.82%	87.47%
F1+F2	74.44%	65.25%	75.75%	81.91%	87.67%
F1+F3	74.54%	63.53%	75.95%	82.82%	88.07%
F2+F3	74.24%	65.35%	75.35%	82.82%	87.87%
F1+F2+F3	75.45%	65.55%	76.15%	83.33%	88.78%

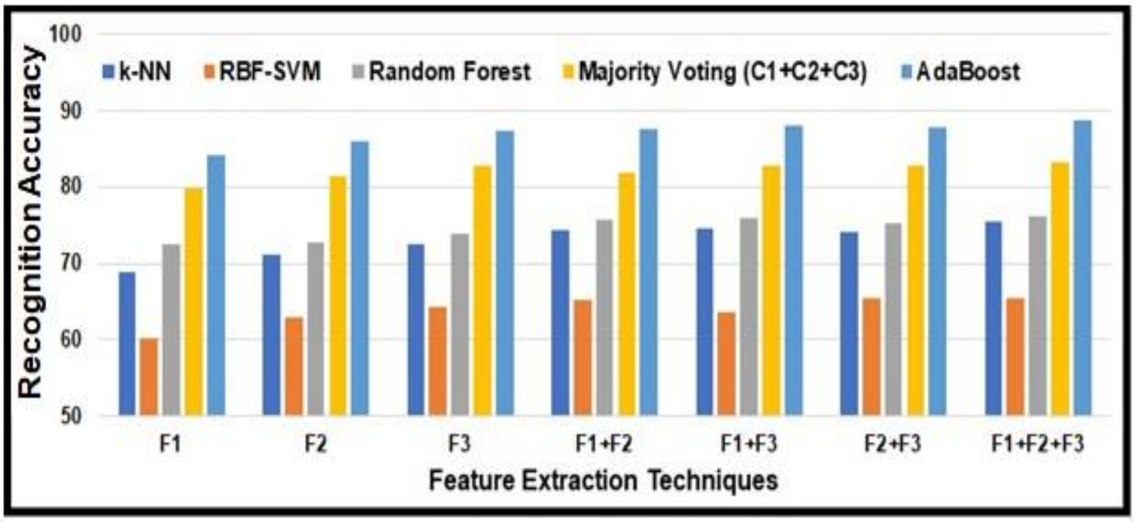

Figure 4.2. System evaluation based on Accuracy

Table 4.2. System evaluation based on FAR

Features	Classification Technique				
	k-NN	RBF-SVM	Random Forest	Majority Voting (C1+C2+C3)	AdaBoost
Zoning (F1)	0.79%	1.29%	0.69%	0.5%	0.5%
Diagonal (F2)	0.69%	1.19%	0.69%	0.5%	0.5%
Intersection and Open-end points (F3)	0.69%	1.09%	0.69%	0.4%	0.4%
F1+F2	0.59%	1.09%	0.69%	0.5%	0.4%
F1+F3	0.59%	1.09%	0.59%	0.4%	0.4%
F2+F3	0.69%	1.09%	0.59%	0.5%	0.4%
F1+F2+F3	0.59%	0.99%	0.59%	0.4%	0.4%

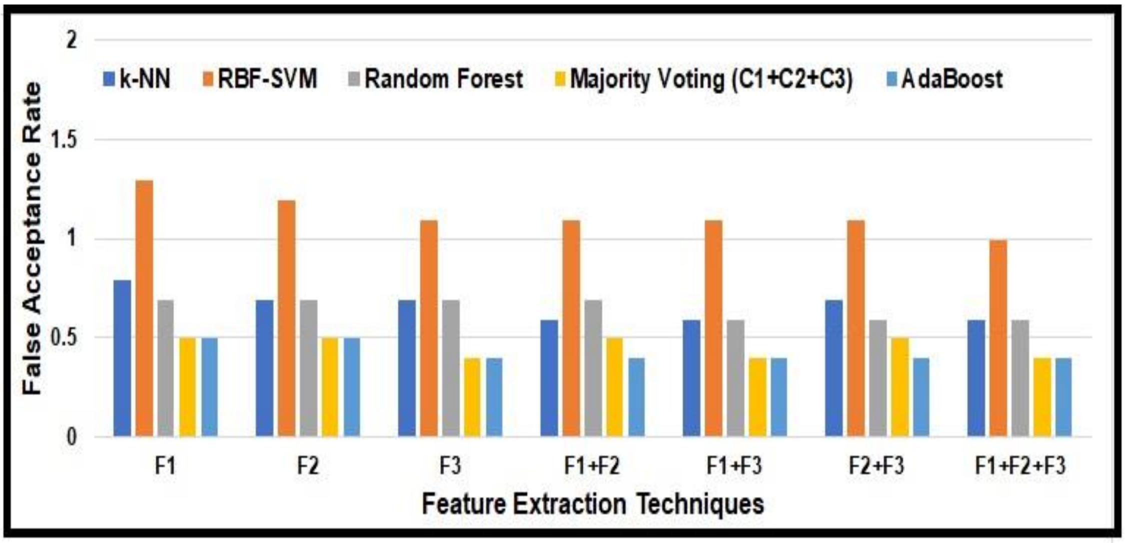

Figure 4.3. System evaluation based on FAR

Table 4.3. System evaluation based on FRR

Features	Classification Technique				
	k-NN	RBF-SVM	Random Forest	Majority Voting (C1+C2+C3)	AdaBoost
Zoning (F1)	30.43%	38.41%	26.69%	19.71%	15.27%
Diagonal (F2)	28.10%	35.89%	26.49%	17.99%	13.55%
Intersection and Open-end points(F3)	26.69%	34.57%	25.38%	16.78%	12.13%
F1+F2	24.97%	33.66%	23.56%	17.59%	11.93%
F1+F3	24.87%	35.38%	23.46%	16.78%	11.53%
F2+F3	25.07%	33.56%	24.06%	16.68%	11.73%
F1+F2+F3	23.96%	33.46%	23.26%	16.27%	10.82%

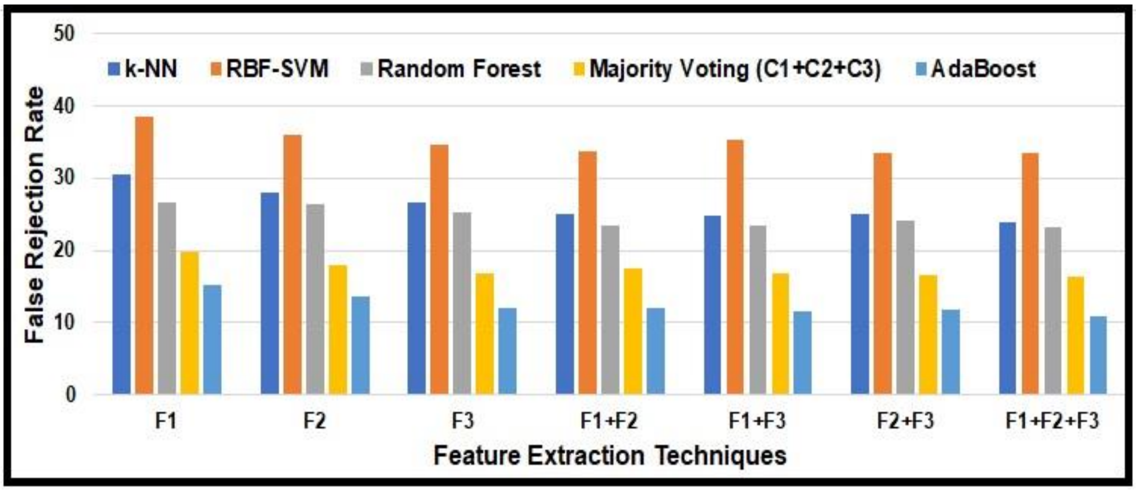

Figure 4.4. System evaluation based on FRR

4.5 COMPARISON WITH THE EXISTING APPROACHES AND SYNTACTIC ANALYSIS

In this section, the comparative analysis of the present approach with state-of-the-art approaches is presented as delineated in Table 4.4.

Table 4.4. Comparison of the present approach with state-of-the-art approaches

Authors	Script	Dataset	Feature Extraction/ Selection Technique	Classification Technique	Accuracy
Kessentini *et al.* (2010)	Arabic and Latin	(i) IFN/ENIT (ii) IRONOFF	Density and contour based features	HMM	(i) 79.8% (ii) 89.8%
Kumar *et al.* (2014a)	Gurumukhi	3500 handwritten character samples	Horizontal peak extent, vertical peak extent, diagonal, and centroid features; Correlation-based Feature Selection (CFS), Principal Component Analysis (PCA) and Consistency-based (CON) feature selection	SVM	91.80% (PCA)
Kumar *et al.* (2016)	Gurumukhi	7000 handwritten character samples	Boundary extent feature extraction; PCA	(i) k-NN (ii) SVM (iii) MLP	93.80% (RBF-SVM)

Authors	**Script**	**Dataset**	**Feature Extraction/ Selection Technique**	**Classification Technique**	**Accuracy**
Assayony and Mahmoud (2017)	Arabic	CENPARMI	Gabor filters integrated with Bag-of-features	SVM	86.44%
Tavoli *et al.* (2018)	Arabic	(i) Iran-cities (ii) IFN/ENIT (iii) IBN SINA	Statistical Geometric Components of Straight lines (SGCSL)	SVM	(i) 67.47% (ii) 80.78% (iii) 86.22%
Arani *et al.* (2019)	Persian	Iranshahr 3	Image gradient, black-white transitions, and contour chain code features	HMM and MLP	89.06%
Present Approach	Gurumukhi	1,00,000 handwritten word samples	Zoning features, diagonal features, intersection and open-end points features	k-NN, RBF-SVM, Random Forest, Majority voting, AdaBoost	88.78% (AdaBoost)

On the basis of experimental results and comparative analysis, we have analyzed the following key points.

- The present approach provided comparable results with the existing approaches applied for word recognition in Arabic, Latin and Persian scripts (Kessentini *et al.*, 2010; Assayony and Mahmoud, 2017; Tavoli *et al.,* 2018; Arani *et al.*, 2019)

- Features play a crucial role in recognition because based on significant characteristics from the word images, the classification task takes place. In the present work, the hybrid of all the three considered features i.e. zoning features, diagonal features and intersection & open-end points features offered significant enhancement in the accuracy.
- In Gurumukhi character recognition, the PCA feature selection technique played a key role to enhance the accuracy by reducing the unnecessary features (Kumar *et al.*, 2014a; Kumar *et al.*, 2016). Thus, it reveals the role of feature selection techniques in classification.
- In the present approach, the Random Forest classifier performed best in comparison to other classifiers as Random Forest amalgamates various Decision Trees in order to enhance the system performance.
- Due to large training samples, the k-NN classifier performed better than the RBF-SVM classifier.
- Majority voting scheme after considering all the three classifiers enhanced the accuracy in comparison to individual base classifiers.
- The present work aimed to explore the AdaBoost methodology and this methodology proved beneficial in order to get a spike in the system performance in terms of accuracy, FAR, and FRR.
- Considering more training data as compared to testing data, the system performance can be enhanced as tested in the present approach using an 80:20 partitioning strategy.

4.6 CHAPTER SUMMARY

In this chapter, we have presented our offline handwritten Gurumukhi word recognition system using a holistic approach. This approach took into account three features such as zoning features, diagonal features, and intersection & open-end points features to extract the desirable characteristics from the words and three classifiers such as k-NN, RBF-SVM, and Random Forest for the classification task. Among all these three classification techniques, the Random Forest classifier offered the best results in terms of accuracy, FAR, and FRR. In order to get significant improvement in system performance, a majority voting scheme and AdaBoost methodology were utilized. Based on the evaluation, the majority voting scheme attained an accuracy of 83.33% using a combination of all three features, and thus boosted the performance of the classifiers. On the other hand, the highest accuracy (88.78%), minimum FAR (0.4%), and minimum FRR (10.82%) were achieved using AdaBoost methodology based on an amalgamation of all three considered features. In terms of accuracy, the present approach is comparable with state-of-the-art approaches in other scripts.