A Novel Framework for Recognizing Characters in Historical Gurmukhi Manuscripts

Harpal Singh[1], Simpel Rani[2], Gurpreet Singh Lehal[3]

[1]Research Scholar, Department of Computer Science, Punjabi University, Patiala, India, Email: harpal.pup@gmail.com
[2]Professor, YCOE, Punjabi University Guru Kashi Campus, Talwandi Sabo, Punjab, India, Email: simpel_jindal@rediffmail.com
[3]Senior Project Consultant, IIIT Hyderabad, Telangana, India, Email: gslehal@pbi.ac.in

Received: 13.07.2024 Revised: 10.08.2024 Accepted: 22.09.2024

ABSTRACT
Ancient manuscripts in the Gurmukhi script serve as invaluable repositories of cultural and historical knowledge, offering insights into the linguistic and artistic heritage of a bygone era. However, the development of efficient Handwritten Text Recognition (HTR) systems for these manuscripts has been hindered by significant challenges. In the past, attempts were made to recognize isolated characters in historical Gurmukhi manuscripts; however, this approach is insufficient for developing a comprehensive HTR system, as these characters mostly appear alongsidevowels. Notably, previous research has not addressed the recognition of characters with vowels. A large dataset is required for the accurate recognition of these characters but due to the scarcity of such datasets, previous attempts to recognize these compounds have not been made. To bridge this gap, we undertook a pioneering effort and meticulously curated 33,223 samples of 156 frequently used character-vowel compounds. To address the complexities of recognition, we extracted various features from these compounds in the collected dataset. Principal Component Analysis (PCA) was applied to the feature set to reduce training time and enhance accuracy. A range of machine learning-based classifiers, such as Support Vector Machine (SVM), Random Forest, k-NN, and Naive Boosting were used for recognition. Additionally, a feature fusion method was used to improve accuracy by combining features from all methods except Open Endpoint features, and PCA was then applied. A notable recognition accuracy of 85.24% was achieved on the features obtained through the feature fusion process with the SVM (RBF) classifier, marking a significant advancement in the recognition of historical Gurmukhi manuscripts and providing a robust foundation for future work in the digital preservation and analysis of literary heritage of India.

Keywords: Historical Gurmukhi Manuscripts, Feature Extraction, Classifiers, Feature Fusion, Machine Learning

1. INTRODUCTION

Gurmukhi script, which is primarily used for writing the Punjabi language, holds a significant place in the cultural and literary heritage of Northern India. Ancient manuscripts written in this script serve as invaluable repositories of historical knowledge, religious texts, and cultural narratives. However, the preservation and accessibility of these manuscripts face significant challenges due to physical degradation and the limited number of scholars capable of deciphering them (Rani 2016). In this context,developing efficient text recognition systems for the Gurmukhi script is crucial for the digital preservation and wider accessibility of these historical documents.

Textrecognition technologies have made significant strides in recent years, particularly for Latin-based scripts. However, the recognition ofthe Gurmukhi script presents unique challenges due to its complex structure and the prevalence of compound characters formed by combining consonants with vowel diacritics. Previous research efforts in Gurmukhi script recognition have primarily focused on isolated character recognition (Rani 2016; Kumar 2019; Singh 2024), which is insufficient for developing a comprehensive HTR system for historical manuscripts.

Recognition of Gurmukhi characters along with vowels is a critical aspect that has been largely overlooked in past studies. This gap in research is primarily due to the scarcity of comprehensive datasets. The lack of such datasets has hindered the development of robust recognition systems capable of handling the intricacies of historical Gurmukhi manuscripts.

The proposed approach addresses this critical gap by introducing a novel dataset and applying machine learning techniques to recognize consonant-vowel compounds in the historical Gurmukhimanuscripts. This paper present a meticulously curated dataset comprising33,223 images of 156 frequently used character-vowel compounds extracted from 22 different historicalGurmukhi manuscripts. The proposed approach significantly expands previous efforts and provides a solid foundation for developing more accurate and comprehensive HTR systems for historical Gurmukhi manuscripts.

To address the complexities of recognizing these compounds, we employ a multi-faceted approach. Our methodology begins with feature extraction, where various attributes are derived from character images to capture distinctive characteristics. To optimize this feature set, we apply Principal Component Analysis (PCA) for dimensionality reduction, which not only reduces training time but also enhances accuracy. For classification, we evaluated the performance of multiple machine learning classifiers, including SVM with RBF and Linear kernels, Random Forest, K-Nearest Neighbors (k-NN), and Naive Boosting. To further improve recognition accuracy, we implement a feature fusion method that combines features extracted using multiple extraction techniques.

The proposed approach not only contributes a valuable dataset to the field but also demonstrates the effectiveness of machine learning techniques in recognizing complex Gurmukhi character-vowel compounds. The results of the proposed approach demonstrate recognition accuracy of 85.24% using an SVM (RBF) classifier on fused features.

1.1 Gurmukhi Script

Gurmukhi scriptwas standardized in the 16th century, specifically for writing Punjabi. It serves as the writing system for Punjabi and the Holy Sikh Scripture Guru Granth Sahib is written in Gurmukhi script. With its alphabetic nature, Gurmukhi incorporates inherent vowel sounds, vowel marks, and characters for nasal sounds. Written from left to right, this script holds profound religious significance for the Sikh community. Gurmukhi is a cursive script comprising forty-one consonants, nine vowels, three sound modifiers (semi-vowels), and three half characters positioned at the feet of the consonants, as illustrated in Figure 1.

In the Gurmukhi script, most characters feature a horizontal line in the upper part. This line, known as the headline, primarily connects the characters within words, resulting in a lack of vertical inter-character gaps.

In terms of structure, a word in Gurmukhi is divided into three horizontal zones, as illustrated in Figure 2. The upper zone hosts vowels above the headline. The middle zone, the most densely populated section, contains consonants and some portions of vowels located below the headline. Below the middle zone lies the lower zone, which accommodates certain vowels and half characters positioned at the base of consonants.

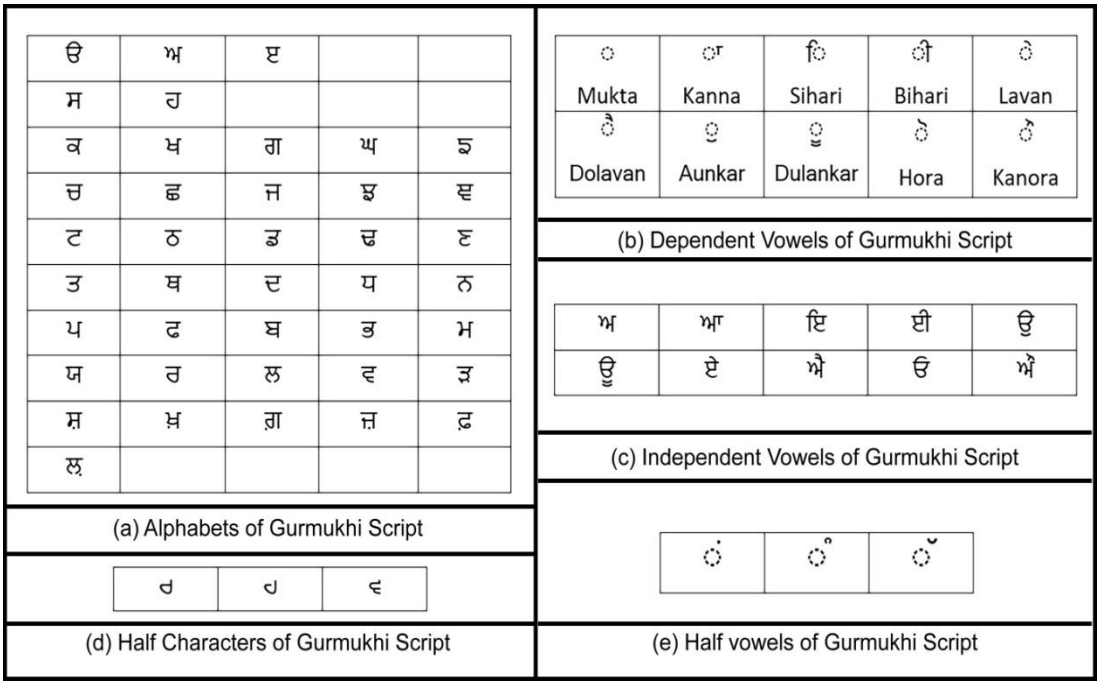

Figure 1: Character set of Gurmukhi script

Figure 2: Different zones of aword in Gumukhi script

The organization of the remaining paper is as follows: Section 2 describes related work. Section 3 presents the dataset development process. The proposed approach is described in Section 4. Experimental results and a discussion are presented in Section 5. Finally, the conclusion is presented in Section 6.

2. RELATED WORK

In recent years, substantial efforts have been made to recognizehandwritten characters in various scripts. However, the recognition of characters in historical manuscripts remains relatively underexplored and only a few authors have attempted to tackle this challenging task. This gap is especially evident in the context of scripts like Gurmukhi where historical manuscripts present unique challenges such as degradation, intricate writing styles and complex structure. On the other hand, some authors have made significant strides in recognizing compound characters found in other scripts. Table 1 presents a comparative analysis of existing character recognition methods for various Indic scripts.

Table 1: Comparative analysis of existing handwritten character recognition methods

Authors	Script	No. of classes	Dataset Size	Feature Extract Method	Classification Method	Accuracy
Rani, 2016	Gurmukhi	35	5600	Discrete Cosine Transformations, Zoning, Gradient and Gabor Filter	SVM (Linear Kernel)	96.21%
Kumar et al., 2019	Gurmukhi	35	1140	Zoning, Discrete Cosine Transformations and gradient features	k-NN, SVM, Decision Tree, Random Forest (adaptive boosting technique)	95.91%
Singh et al., 2024	Gurmukhi	33	87181	CNN	CNN	99.77%
Narasimhaiah, S. T.,et al, 2022	Kannada	300	15,000	CNN	CNN	98.80%
Shelke, S. et al ,2011	Marathi	300	35000	Wavelet Decomposition and Modified Wavelet Features	Neural Network	94.22%
Sachdeva, J. et al. ,2021	Devanagari	20	5000	Edge Histogram,	SVM	99.88%
Kadam A.A. et al ,2019	Marathi	35	3500	Zoning and Statistical feature	SVM, k-NN	96.49%, 95.67%
Islam, M.S. et al ,2022	Bengali	256	17049	CNN	CNN	93.42%

Sayeed, A. et al ,2021	Bengali	256	11,935	CNN	CNN	96%
Pramanik, R. et al. ,2018	Bangla	128	10,240	Chain Code Histogram	MLP	80.50%
Das, N. et al, 2010	Bangla	210	25000	Shadow Features, Longest Run Features	MLP and SVM	79.25% 80.510%
Roy, S. et al,2017	Bangla	171	34200	DCNN	DCNN	90.33%
Pal, U. et al, 2007	Bangla	392	20,543	Gradient Feature	MQDF	85.90%
Ashiquzzaman, A. et al, 2017	Bangla	171	34,000	DCNN	DCNN	93%
Chakraborty, S. et al, 2021	Bengali	122	300,00	CNN	CNN	89.20%
Das, N. et al, 2014	Bangla	199	55,278	Convex Hull, Quadtree-Based Feature Set	SVM	79.35%
Garain U. et al, 1998	Bangla	250	25000	Number-based Metric Distance	SVM	99.69%.
Muppalaneni, N.B. et al,2020	Telugu	16	5160	CNN	CNN	79.61%
Sharif, S. M. A. et al,2018	Bangla	199	44,152	CNN	CNN	92.77%

3. Dataset Development

3.1 Dataset Collection

We visited various libraries and collected 1356 pages from22 different historical Gurmukhi scripts. The distribution of the samples in the collected dataset is given in Table 2:

Table 2: Distribution of character samples in the collected dataset

Character	No. Samples	Character	No. Samples	Character	No. Samples	Character	No. Samples
ੳ	395	ਗੋ	138	ਤੁ	110	ਭੁ	114
ਉ	101	ਗੇ	126	ਤੇ	395	ਭੂ	102
ਐ	102	ਗਾ	216	ਤਾ	556	ਭਾ	207
ਆ	851	ਗਿ	182	ਤਿ	881	ਯੰ	146
ਅੰ	261	ਗੀ	122	ਤੀ	292	ਭੀ	159
ਏ	262	ਘਾ	113	ਤੰ	152	ਮੁ	165
ਇ	947	ਚੁ	112	ਥਾ	102	ਮੂ	110
ਈ	367	ਚੰ	102	ਦੁ	106	ਮੋ	105
ਸੁ	454	ਚਾ	216	ਦੂ	122	ਮੇ	125
ਸੂ	124	ਚਿ	137	ਦੋ	188	ਮੈ	203
ਸੋ	169	ਚੀ	158	ਦੇ	337	ਮਾ	491
ਸੇ	140	ਚੰ	130	ਦਾ	218	ਮਿ	179
ਸਾ	311	ਛੁ	140	ਦਿ	205	ਮੀ	112
ਸਿ	264	ਛਾ	125	ਦੀ	113	ਮੰ	237
ਸੀ	149	ਛੰ	140	ਧੁ	104	ਯੋ	193
ਸੰ	261	ਜੁ	104	ਧਾ	149	ਯੇ	100
ਹੁ	143	ਜੋ	151	ਧਿ	131	ਯਾ	217
ਹੋ	243	ਜੇ	123	ਨੁ	104	ਰੁ	112
ਹੇ	115	ਜੈ	113	ਨੋ	170	ਰੂ	122
ਹੈ	274	ਜਾ	410	ਨੇ	135	ਰੋ	133
ਹਾ	268	ਜਿ	166	ਨੈ	168	ਰੇ	317
ਹਿ	852	ਜੀ	250	ਨਾ	695	ਰੈ	218

ਹੀ	343	ਝ	118	ਨਿ	591	ਰਾ	717
ਹੰ	119	ਝੈ	123	ਨੀ	254	ਰਿ	473
ਕ	121	ਝਾ	145	ਨੰ	141	ਰੀ	256
ਕੋ	216	ਟ	110	ਪੁ	264	ਰੰ	110
ਕੇ	280	ਟੇ	150	ਪੂ	144	ਲੋ	108
ਕੈ	124	ਟਾ	122	ਪੈ	102	ਲੇ	147
ਕਾ	644	ਟਿ	140	ਪਾ	488	ਲੈ	140
ਕਿ	217	ਟੀ	157	ਪਿ	121	ਲਾ	272
ਕੀ	589	ਠਾ	132	ਪੰ	125	ਲਿ	148
ਕੰ	104	ਡਾ	120	ਫ਼	105	ਲੀ	160
ਖ	106	ਡਿ	110	ਫਿ	102	ਵੇ	113
ਖੇ	103	ਢਾ	136	ਬੁ	102	ਵੈ	324
ਖੈ	116	ਣ	155	ਬੇ	97	ਵਾ	402
ਖਾ	153	ਣੇ	121	ਬੈ	121	ਵਿ	372
ਖਿ	114	ਣਾ	110	ਬਾ	161	ਵੀ	102
ਖੀ	122	ਣੀ	101	ਬਿ	254	ਸ਼ੈ	127
ਗੁ	314	ਤ	211	ਬੀ	104	ਸ਼ੀ	103

3.2 Challenges in the collected dataset

After developing the dataset, we found that the collected data presented a diverse range of challenges. The diverse challenges of the dataset are represented in Figure 3. These challenges encompassed:

- The authors used various types of pencils and ink, leading to noticeable variations in the line thickness.
- Fragmented and broken characters were prevalent, introducing complexity in the recognition process.
- The presence of noise and degraded image quality pose additional challenges in the recognition process. The use of multicolored ink and different writing styles further complicated the process.
- The use of different tools for cropping introduces variability in image quality and size, which further complicates recognition process.

In addition to these challenges, variations in the shape of characters are clearly depicted in Figure 3.

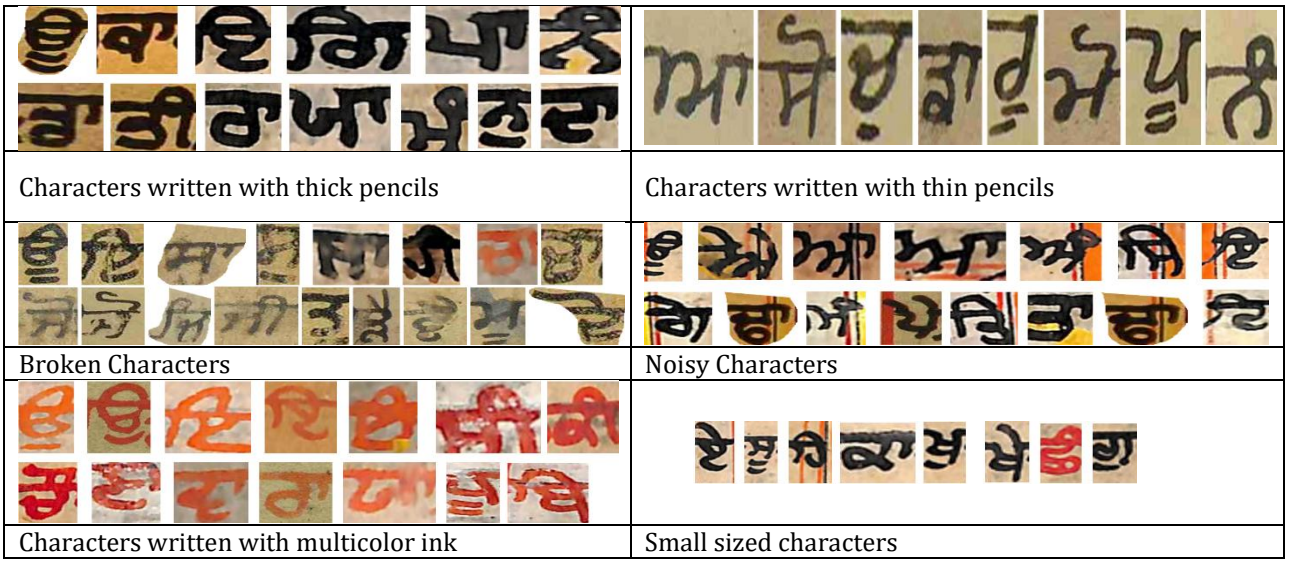

Figure 3: Challenges in collected dataset

4. Proposed Approach

The proposed approach comprises several stages, including image acquisition, character segmentation, preprocessing, and recognition, as depicted in Figure 4. These stages are briefly outlined in the subsequent section.

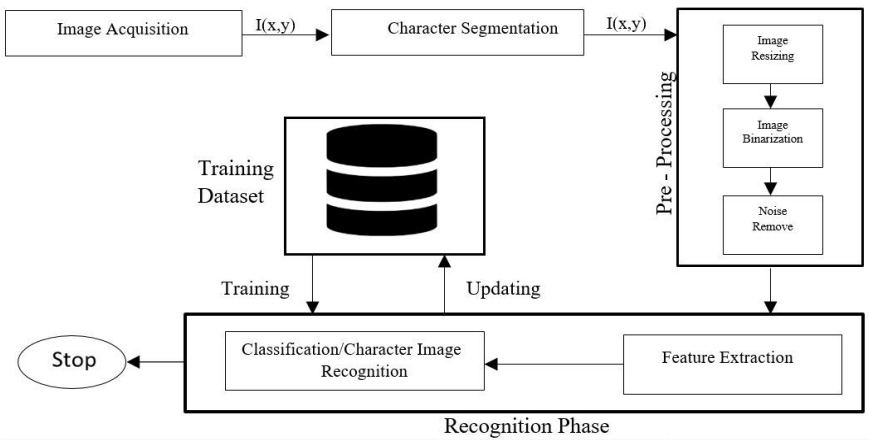

Figure 4: Proposed approach

4.1 Character Segmentation

We manually cropped 33223 samples of character-vowel compounds using different cropping tools from 1356 pages of 22 different Gurmukhi manuscripts. A few samples of the cropped dataset are shown in Figure 5.

4.2 Sample resizing

Due to the diverse sources of manuscripts and cropping tools used to extract images of character-vowel compounds, the dimensions of the cropped images exhibited variability. To simplify the feature extraction process, all image samples were resized to a standardized dimension of 64×64 pixels using nearest neighbor interpolation.

4.3 Image Binarization and Noise Removal

The experiments were conducted using various binarization methods, including the adaptive binarization method, Sauvolathresholding, Niblackthresholding, and Otsu's thresholding method to convert the colored images into binary images. The results ofSauvola'sthresholding method were found to be better than those of other thresholdingmethods. Therefore, we used Sauvola'sthresholding method for binarization. Morphological operations were employed to effectively remove various types of noise in binary images.

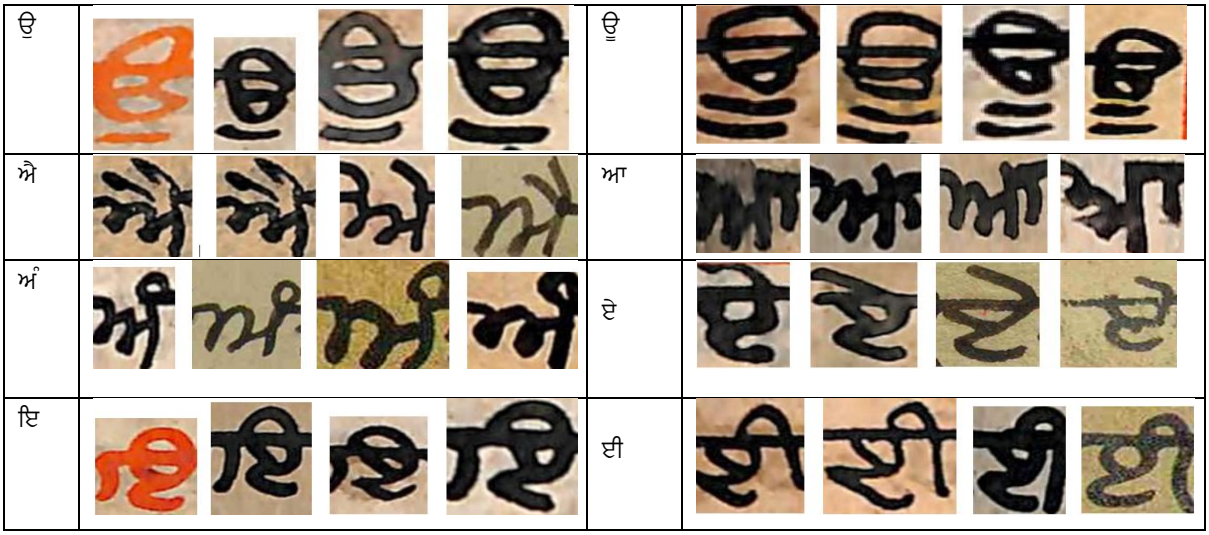

Figure 5: Sample images of collected dataset

4.4 Feature Extraction

Feature extraction plays a crucial role in character recognition and classification by extracting meaningful information from character images. The accuracy the recognition process significantly depends on the selection of an appropriate feature set. We adopted the Hierarchical Zoning method for feature extraction, as shown in Figure 6.Initially, the entire character image is treated as a single zone, and features are

computed for this zone. Subsequently, the image was divided into four equally sized zones. This process continues with subdivision of the four zones into sixteen zones and feature computation for each subdivision. Further subdivision results in sixty-four zones, each sized 8 x 8 pixels, with features extracted for each zone. Consequently, we obtain a feature vector encompassing 85 zones (1 + 4 + 16 + 64).

(a) (b) (c) (d)

Figure 6: Character Images: (a) one zone (b) four zones (c) sixteen zones (d) sixty four zones

Feature extraction was performed using the following methods:

4.1.1 Diagonal features

In a zone consisting of n rows and m columns, there are a total of n + m - 1 diagonals. To extract features using the diagonal method, the system calculates the sum of black pixels along each diagonal in the zone and stores these sums in a list. Subsequently, the values in the list are averaged to generate a single feature. This procedure is applied iteratively for each zoneto ensure comprehensive feature extraction across the image (Pradeep, J., 2011).

4.1.2 Zonal Features

The zonal feature extraction method involves summing the number of black pixels in each zone to create a single feature (Pradeep, J., Srinivasan, E. and Himavathi, S.,2011).

4.1.3 Horizontal and Vertical Peak Extent Features

The Vertical Peak Extent feature assesses the maximum vertical extent of consecutive black pixels in each column of a zone, treating it as a single feature. Similarly, the Horizontal Peak Extent feature evaluates the maximum horizontal extent of consecutive black pixels in each row of a zone and considers it as a single feature (Kumar, M., 2018).

4.1.4 Intersection and Open Endpoint Features

An intersection point is defined as a black pixel surrounded by more than one black pixel in its neighborhood, while an open endpoint refers to a black pixel with only one black pixel in its neighborhood. To extract Intersection and Open Endpoint Features, the system calculates the number of intersections and Open Endpoints in each zone (Arora, S.,2008).

4.5 Classification

Classification is the process of categorizingdata into predefined classes or categories based on the input features. Through the use of supervised learning algorithms, classification models learn patterns from labeled training data to predictthe class labels of unseen data instances. During this phase, the model categorizes unknown character images. In the experimental phase, we explored various classifiers, including Random Forest, SVM with Linear and RBF kernels, Naive Bayes, and k-NN, to classify characters. These classifiers were applied to different sets of features and their combinations. The features extracted in the preceding step were used for classification.

5. Experimental Results And Discussion

In this section, we present the experimental results obtained after employing the SVM (RBF) (C1), SVM (Linear) (C2), Random Forest (C3), k-NN (C4) and Naive Boosting (C5) classifiers on the features extracted using Diagonal Features (F1), Zonal Features (F2), Horizontal Peak Extent Features (F3), Vertical Peak Extent Features (F4), Intersection Features (F5) and Open Endpoint Features (F6) from the collected dataset of 33223 images of 156 frequently used character-vowel compounds as mentioned in the section 5.5. These classifiers underwent training and testing using both individual and combined feature sets. The feature sets were randomly split into training and testing sets with 70% of the features from feature sets allocated for training and the remaining 30% allocated for testing.

Initially, the experiments were conducted with all individual features (F1 – F6) and all classifiers (C1-C5). The SVM classifier with the RBF kernel achieved the highest recall rate of 85.04% on the diagonal features. The k-NN classifier consistently demonstrated good accuracy across all features. The accuracy of

the Random Forest classifier appeared to be better than the remaining classifiers across all features. The results obtained using different classifiers on these features are shown in Figure 7 and Table 3.

Table 3: Recognition accuracy of different classifiers on individual feature sets

Feature Extraction Method	SVM (RBF)	SVM (Linear)	Random Forest	k-NN	Naive Bayes
Diagonal Features	0.8504	0.7925	0.6270	0.7349	0.4894
Zonal Features	0.0304	0.3279	0.6199	0.6603	0.4746
Horizontal Peak Extent Features	0.0515	0.4212	0.6370	0.7123	0.4779
Vertical Peak Extent Features	0.0304	0.4623	0.6367	0.7349	0.5275
Intersection Point Features	0.3078	0.2929	0.2397	0.1418	0.1392
Open Endpoint Features	0.0304	0.3300	0.6200	0.6650	0.4735

We applied PCA to select the important features from the feature set and reduce training time. They reduced the number of features from 85 to 63 using PCA. The classification accuracies of the different classifiers after applying PCA are shown in Table 4 and Figure 8.

Table 4: Recognition accuracy of different classifiers on individual feature sets using PCA

Feature Extraction Methods	SVM (RBF)	SVM (Linear)	Random Forest	k-NN	Naive Bayes
Diagonal Features	0.8371	0.7926	0.5283	0.7354	0.6372
Zonal Features	0.0304	0.3278	0.5331	0.6603	0.6066
Horizontal Peak Extent Features	0.0733	0.4213	0.5544	0.7113	0.6236
Vertical Peak Extent Features	0.0306	0.4609	0.5583	0.7346	0.6711
Intersection Point Features	0.0304	0.3298	0.5228	0.6650	0.6090
Open Endpoint Features	0.3017	0.2446	0.2144	0.1375	0.1805

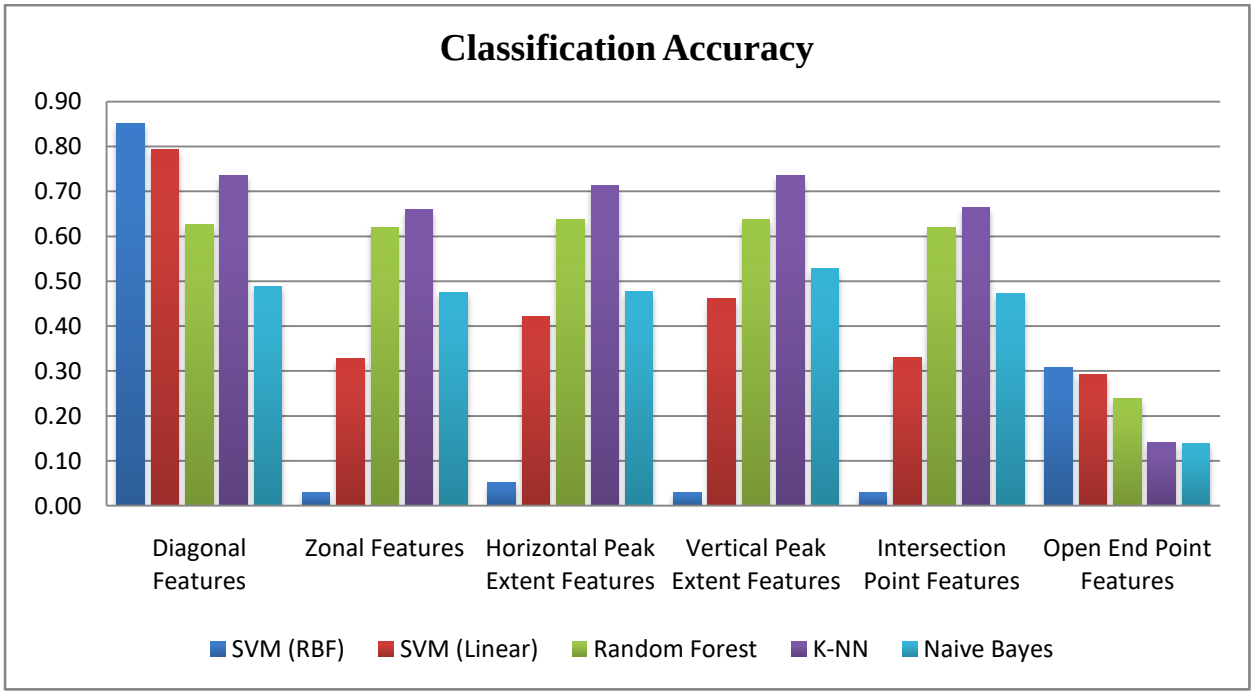

Figure 7: Recognition accuracy of different classifiers on individual feature sets

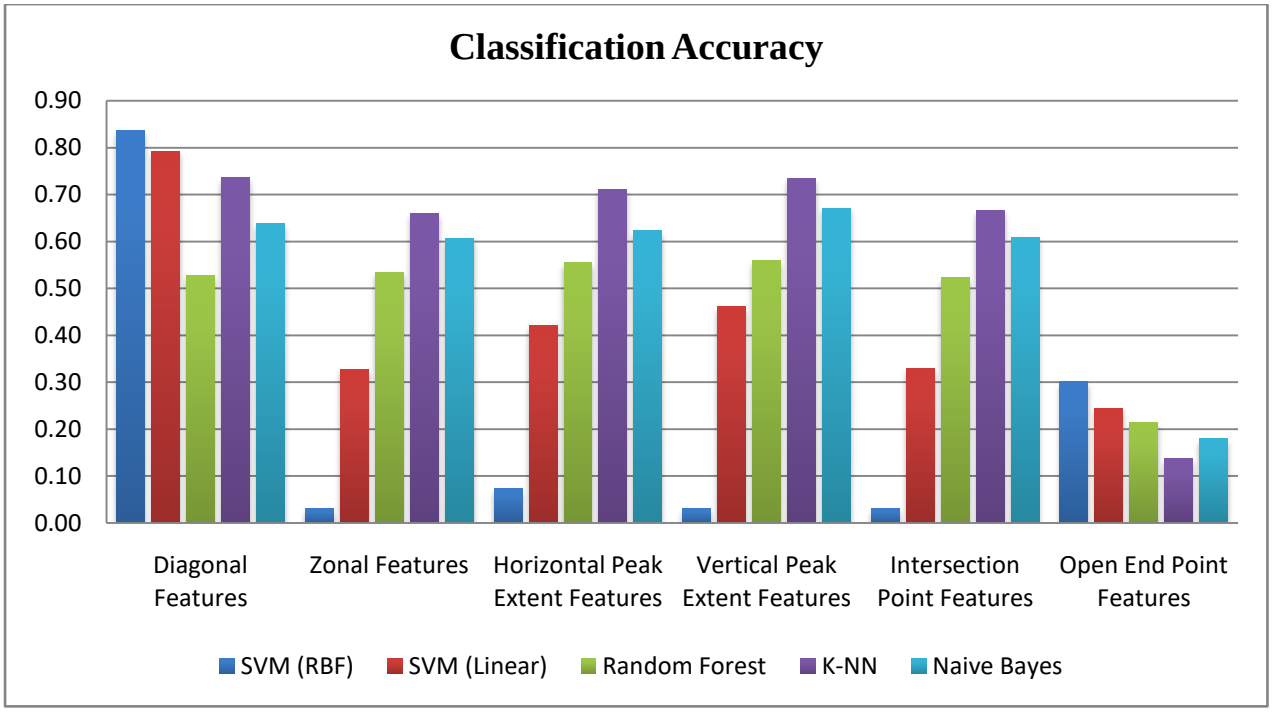

Figure 8: Recognition accuracy of different classifiers on individual feature sets using PCA

The recognition accuracy of all classifiers was lower when using Open Endpoint features. Therefore, Open Endpoint features were excluded from further experiments. Thereafter, we generated all possible unique combinations of the two features and applied all classifiers to these combinations. This process was repeated for combinations of three features and then four features, each time applying all classifiers to the new combinations. Finally, all five feature sets were combined into a singlecomprehensive feature set and all classifiers were applied to this set. To assess the performance of classifiers on different feature sets, the precision, recall, and F1 score metrics were used. The performance of the classifiers on different feature sets is depicted in Figure 9 and Table 5.

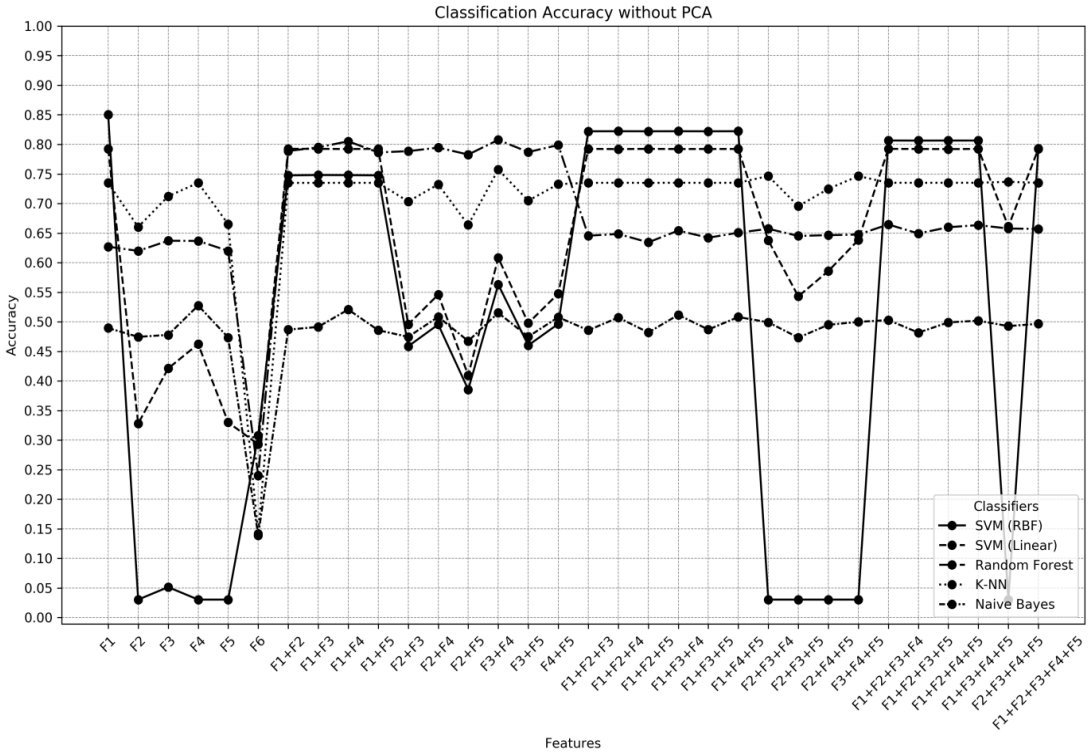

Figure 9: Recognition accuracy of different classifiers on individual and combined feature sets

To improve the accuracy, PCAwas applied to the combined feature sets. The results obtained from different classifiers on the reduced feature set are illustrated in Figure 10 and Table 6. Notably, the SVM classifier with an RBF kernel achieved the highest recall rate of 85.24%using the combined features from all feature extraction methods.

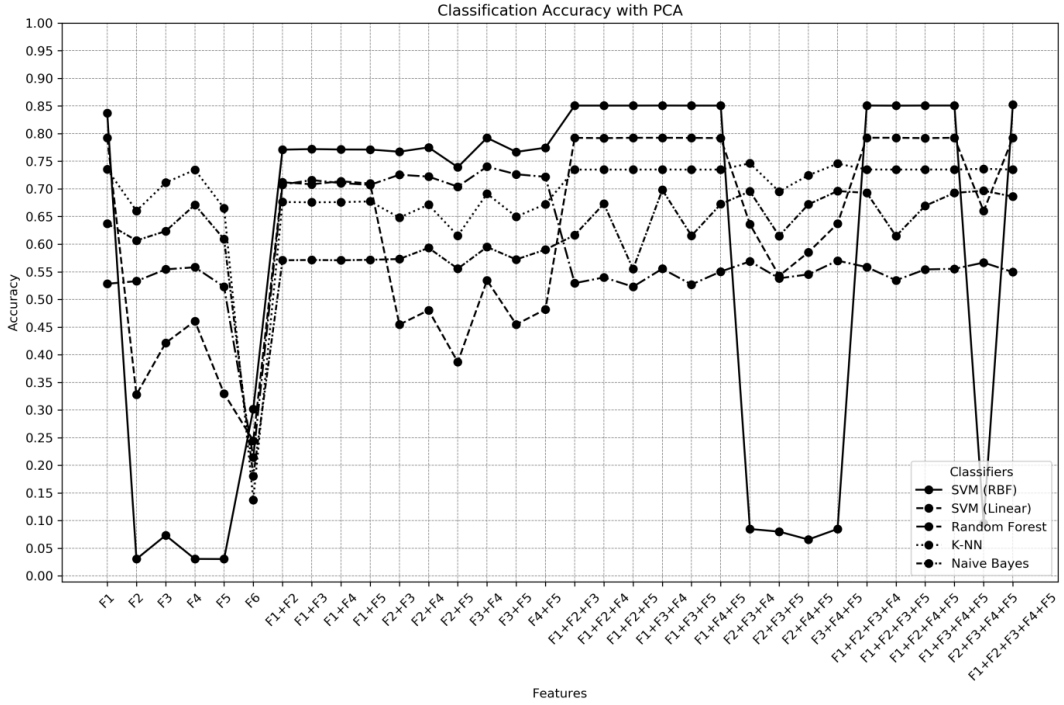

Figure 10: Recognition accuracy of different classifiers on individual and combined feature sets using PCA

6. CONCLUSION AND FUTUREWORK

This paper attempts to recognize character-vowel compounds in historicalGurmukhi manuscripts; unveiling formidable challenges that encompass noise, rust, degraded quality, multicolor ink, and fragmented characters. In response to the scarcity of datasets, we meticulously curated 33,223 samples from 156 frequently used compound characters from 22 historicalGurmukhi manuscripts. The proposed approach is the first attempt to create a specialized dataset of 156 frequently used character-vowel compounds in historicalGurmukhi manuscripts. The recognition approach extracts diverse features from the curated dataset. This is followed by the application of PCA for feature selection and a variety of machine learning classifiers, including SVM (RBF), SVM (Linear), Random Forest, k-NN, and Naive Boosting. The proposed model achieved a notable recall rate of 85.24%, particularly when the SVM (RBF) classifier was trained on the refined features obtained through the fusion process.

In the future, we plan to expand the dataset and explore deep learning models to improve recognition accuracy.

Table 5: Recognition accuracy of different classifiers on individual and combined feature sets

	SVM (RBF)			SVM (Linear)			Random Forest			k-NN			Naive Bayes		
Method	**Precision**	**Recall**	**F1-Score**	**Precision**	**Recall**	**F1-Score**	**Precision**	**Recall**	**F1-Score**	**Precision**	**Recall**	**F1-Score**	**Precision**	**Recall**	**F1-Score**
F1	0.8573	0.8504	0.8486	0.7968	0.7925	0.7915	0.6438	0.6270	0.6173	0.7445	0.7349	0.7310	0.5663	0.4894	0.4975
F2	0.0009	0.0304	0.0018	0.2142	0.3279	0.2233	0.6351	0.6199	0.6105	0.6764	0.6603	0.6558	0.5500	0.4746	0.4826
F3	0.0195	0.0515	0.0206	0.3465	0.4212	0.3270	0.6488	0.6370	0.6272	0.7238	0.7123	0.7091	0.5564	0.4779	0.4853
F4	0.000	0.03	0.00	0.39	0.46	0.37	0.65	0.63	0.62	0.74	0.73	0.73	0.58	0.52	0.53

	9	04	18	18	23	16	18	67	69	73	49	22	94	75	64
F5	0.0009	0.0304	0.0018	0.2115	0.3300	0.2252	0.6356	0.6200	0.6097	0.6799	0.6650	0.6601	0.5475	0.4735	0.4809
F6	0.3594	0.3078	0.2703	0.2879	0.2929	0.2838	0.3027	0.2397	0.2361	0.1444	0.1418	0.1223	0.2107	0.1392	0.1419
F1+F2	0.7611	0.7475	0.7403	0.7970	0.7925	0.7916	0.7970	0.7888	0.7825	0.7445	0.7349	0.7310	0.5665	0.4867	0.4953
F1+F3	0.7642	0.7481	0.7410	0.7967	0.7924	0.7914	0.8037	0.7949	0.7885	0.7445	0.7349	0.7310	0.5749	0.4914	0.5007
F1+F4	0.7639	0.7479	0.7408	0.7965	0.7921	0.7912	0.8128	0.8051	0.7994	0.7446	0.7350	0.7311	0.5964	0.5207	0.5314
F1+F5	0.7611	0.7475	0.7403	0.7968	0.7924	0.7915	0.7948	0.7864	0.7794	0.7445	0.7349	0.7310	0.5652	0.4858	0.4943
F2+F3	0.3886	0.4586	0.3722	0.4478	0.4956	0.4188	0.7934	0.7885	0.7813	0.7175	0.7034	0.6998	0.5542	0.4746	0.4832
F2+F4	0.4553	0.4954	0.4131	0.5149	0.5460	0.4807	0.8025	0.7945	0.7884	0.7460	0.7324	0.7300	0.5823	0.5084	0.5184
F2+F5	0.2853	0.3854	0.2844	0.3297	0.4095	0.3162	0.7898	0.7827	0.7762	0.6798	0.6637	0.6593	0.5501	0.4675	0.4772
F3+F4	0.5201	0.5630	0.4927	0.6039	0.6082	0.5570	0.8149	0.8075	0.8021	0.7697	0.7574	0.7552	0.5883	0.5152	0.5244
F3+F5	0.3909	0.4602	0.3741	0.4440	0.4977	0.4215	0.7929	0.7870	0.7807	0.7185	0.7048	0.7012	0.5561	0.4749	0.4837
F4+F5	0.4553	0.4964	0.4144	0.5208	0.5476	0.4825	0.8064	0.7988	0.7930	0.7462	0.7328	0.7303	0.5832	0.5078	0.5183
F1+F2+F3	0.8301	0.8221	0.8196	0.7968	0.7924	0.7914	0.6619	0.6456	0.6368	0.7445	0.7349	0.7310	0.5671	0.4857	0.4950
F1+F2+F4	0.8302	0.8222	0.8196	0.7964	0.7920	0.7911	0.6617	0.6485	0.6396	0.7446	0.7351	0.7312	0.5837	0.5072	0.5180
F1+F2+F5	0.8301	0.8221	0.8196	0.7969	0.7924	0.7915	0.6487	0.6343	0.6250	0.7445	0.7349	0.7310	0.5629	0.4820	0.4911
F1+F3+F4	0.8302	0.8222	0.8197	0.7965	0.7921	0.7912	0.6709	0.6542	0.6461	0.7446	0.7351	0.7312	0.5871	0.5117	0.5220
F1+F3+F5	0.8301	0.8221	0.8196	0.7965	0.7922	0.7912	0.6576	0.6421	0.6345	0.7445	0.7349	0.7310	0.5693	0.4869	0.4965
F1+F4+F5	0.8302	0.8222	0.8196	0.7966	0.7922	0.7913	0.6688	0.6507	0.6433	0.7446	0.7351	0.7312	0.5851	0.5081	0.5190
F2+F3+F4	0.0009	0.0304	0.0018	0.6507	0.6375	0.5990	0.6728	0.6572	0.6492	0.7589	0.7464	0.7439	0.5760	0.4991	0.5090
F2+F3+F5	0.0009	0.0304	0.0018	0.5234	0.5430	0.4822	0.6573	0.6452	0.6359	0.7107	0.6954	0.6916	0.5553	0.4733	0.4825
F2+F4+F5	0.0009	0.0304	0.0018	0.5995	0.5858	0.5350	0.6642	0.6465	0.6382	0.7377	0.7247	0.7219	0.5759	0.4949	0.5058
F3+F4+F5	0.0009	0.0304	0.0018	0.6501	0.6382	0.6000	0.6619	0.6476	0.6387	0.7585	0.7463	0.7438	0.5769	0.4998	0.5098
F1+F2+F3+F4	0.8159	0.8065	0.8036	0.7968	0.7924	0.7915	0.6837	0.6645	0.6573	0.7446	0.7351	0.7312	0.5820	0.5030	0.5134
F1+F2+F3+F5	0.8157	0.8063	0.8034	0.7965	0.7922	0.7913	0.6634	0.6493	0.6411	0.7446	0.7350	0.7311	0.5638	0.4817	0.4916
F1+F2+F4+F5	0.8158	0.8064	0.8035	0.7963	0.7920	0.7911	0.6744	0.6599	0.6526	0.7446	0.7351	0.7312	0.5767	0.4990	0.5095
F1+F3+F4+F5	0.8158	0.8064	0.8035	0.7967	0.7923	0.7914	0.6767	0.6634	0.6546	0.7446	0.7351	0.7312	0.5810	0.5019	0.5121
F2+F3+F4+F5	0.0009	0.0304	0.0018	0.6770	0.6610	0.6306	0.6694	0.6577	0.6484	0.7498	0.7367	0.7340	0.5741	0.4927	0.5039

F1+F2+F3+F4+F5	0.8033	0.7927	0.7891	0.7967	0.7924	0.7915	0.6729	0.6570	0.6489	0.7445	0.7350	0.7311	0.5765	0.4966	0.5074

Table 6: Recognition accuracy of different classifiers on individual and combined feature sets using PCA

Method	SVM (RBF)			SVM (Linear)			Random Forest			k-NN			Naive Bayes		
	Precision	Recall	F1-Score	Precision	Recall	F1-Score	Precision	Recall	F1-Score	Precision	Recall	F1-Score	Precision	Recall	F1-Score
F1	0.8482	0.8371	0.8351	0.7967	0.7926	0.7917	0.5639	0.5283	0.5173	0.7452	0.7354	0.7316	0.6774	0.6372	0.6486
F2	0.0009	0.0304	0.0018	0.2141	0.3278	0.2233	0.5681	0.5331	0.5223	0.6764	0.6603	0.6558	0.6434	0.6066	0.6163
F3	0.0320	0.0733	0.0196	0.3467	0.4213	0.3271	0.5823	0.5544	0.5436	0.7233	0.7113	0.7082	0.6617	0.6236	0.6319
F4	0.0082	0.0306	0.0021	0.3941	0.4609	0.3699	0.5958	0.5583	0.5476	0.7469	0.7346	0.7319	0.7099	0.6711	0.6824
F5	0.0009	0.0304	0.0018	0.2114	0.3298	0.2251	0.5549	0.5228	0.5113	0.6799	0.6650	0.6601	0.6440	0.6090	0.6182
F6	0.3631	0.3017	0.2669	0.2356	0.2446	0.2287	0.2973	0.2144	0.2179	0.1360	0.1375	0.1190	0.1819	0.1805	0.1721
F1+F2	0.7776	0.7710	0.7664	0.7154	0.7120	0.7098	0.7200	0.7079	0.6958	0.6852	0.6763	0.6703	0.5966	0.5709	0.5767
F1+F3	0.7788	0.7720	0.7673	0.7127	0.7084	0.7063	0.7255	0.7160	0.7037	0.6855	0.6758	0.6700	0.5972	0.5714	0.5773
F1+F4	0.7780	0.7713	0.7667	0.7174	0.7138	0.7114	0.7195	0.7105	0.6986	0.6861	0.6761	0.6704	0.5959	0.5709	0.5766
F1+F5	0.7775	0.7712	0.7666	0.7126	0.7098	0.7073	0.7154	0.7073	0.6951	0.6865	0.6774	0.6715	0.5969	0.5717	0.5774
F2+F3	0.7731	0.7673	0.7626	0.3921	0.4545	0.3726	0.7337	0.7256	0.7160	0.6603	0.6481	0.6430	0.5925	0.5731	0.5742
F2+F4	0.7809	0.7751	0.7713	0.4436	0.4806	0.4034	0.7284	0.7222	0.7117	0.6822	0.6715	0.6661	0.6146	0.5933	0.5970
F2+F5	0.7471	0.7393	0.7342	0.2936	0.3867	0.2904	0.7138	0.7041	0.6934	0.6313	0.6153	0.6101	0.5771	0.5556	0.5586
F3+F4	0.7977	0.7924	0.7891	0.5096	0.5346	0.4687	0.7469	0.7402	0.7316	0.7025	0.6913	0.6869	0.6173	0.5949	0.5970
F3+F5	0.7730	0.7669	0.7623	0.3911	0.4548	0.3729	0.7340	0.7264	0.7163	0.6618	0.6498	0.6447	0.5923	0.5723	0.5737
F4+F5	0.7802	0.7745	0.7707	0.4463	0.4821	0.4054	0.7308	0.7218	0.7126	0.6824	0.6724	0.6668	0.6122	0.5902	0.5945
F1+F2+F3	0.8574	0.8507	0.8489	0.7968	0.7924	0.7914	0.5665	0.5295	0.5170	0.7445	0.7349	0.7310	0.6651	0.6164	0.6272
F1+F2+F4	0.8574	0.8507	0.8489	0.7963	0.7919	0.7910	0.5828	0.5399	0.5308	0.7446	0.7350	0.7311	0.7164	0.6731	0.6856
F1+F2+F5	0.8574	0.8507	0.8489	0.7970	0.7925	0.7916	0.5582	0.5230	0.5123	0.7445	0.7349	0.7310	0.6279	0.5549	0.5716
F1+F3+F4	0.8574	0.8508	0.8490	0.7969	0.7925	0.7916	0.5869	0.5555	0.5420	0.7446	0.7351	0.7312	0.7375	0.6982	0.7101
F1+F3+F5	0.8574	0.8507	0.8489	0.7966	0.7923	0.7913	0.5677	0.5270	0.5160	0.7445	0.7349	0.7310	0.6643	0.6153	0.6262
F1+F4+F5	0.8573	0.8507	0.8489	0.7964	0.7920	0.7911	0.5913	0.5504	0.5411	0.7446	0.7350	0.7311	0.7158	0.6725	0.6851
F2+F3+F4	0.0254	0.0848	0.0261	0.6531	0.6360	0.5971	0.6018	0.5688	0.5589	0.7584	0.7463	0.7439	0.7334	0.6955	0.7062
F2+F3+F5	0.0242	0.0800	0.0235	0.5197	0.5427	0.4818	0.5676	0.5383	0.5256	0.7104	0.6952	0.6913	0.6545	0.6149	0.6247

F2+F4+F5	0.0069	0.0657	0.0121	0.5965	0.5852	0.5344	0.5833	0.5453	0.5345	0.7373	0.7246	0.7216	0.7125	0.6720	0.6829
F3+F4+F5	0.0258	0.0848	0.0263	0.6505	0.6371	0.5985	0.6063	0.5702	0.5608	0.7580	0.7457	0.7432	0.7343	0.6961	0.7068
F1+F2+F3+F4	0.8574	0.8508	0.8490	0.7969	0.7926	0.7917	0.5953	0.5585	0.5486	0.7446	0.7351	0.7312	0.7339	0.6927	0.7050
F1+F2+F3+F5	0.8573	0.8506	0.8488	0.7968	0.7925	0.7915	0.5657	0.5347	0.5217	0.7446	0.7350	0.7311	0.6660	0.6147	0.6267
F1+F2+F4+F5	0.8574	0.8508	0.8490	0.7963	0.7919	0.7910	0.5957	0.5543	0.5443	0.7446	0.7351	0.7312	0.7142	0.6694	0.6827
F1+F3+F4+F5	0.8574	0.8508	0.8490	0.7969	0.7925	0.7916	0.5947	0.5554	0.5438	0.7446	0.7351	0.7312	0.7339	0.6927	0.7049
F2+F3+F4+F5	0.0433	0.0951	0.0339	0.6767	0.6599	0.6294	0.6002	0.5667	0.5566	0.7492	0.7363	0.7336	0.7340	0.6967	0.7072
F1+F2+F3+F4+F5	0.8587	**0.8524**	0.8506	0.7969	0.7926	0.7917	0.5898	0.5495	0.5404	0.7445	0.7350	0.7311	0.7303	0.6861	0.6994

REFERENCES

[1] Rani, S. (2016). Recognition of Handwritten Gurmukhi Manuscripts.Doctoral dissertation, Punjabi University, Patiala, Punjab, India. pp. 1-170.

[2] Singh, H., Rani, S. andLehal, G.S.(2024). An Efficient Transfer Learning Approach for Handwritten Historical GurmukhiCharacter Recognition using VGG16: Gurmukhi_Hhdb1.0.Library Progress International,44(1s), 103-114.

[3] Kumar, M., Jindal, S.R., Jindal, M. K., &Lehal, G.S. (2019).Improved recognition results of medieval handwritten Gurmukhi manuscripts using boosting and bagging methodologies.Neural Processing Letters,50, 43-56.

[4] Narasimhaiah, S. T., Rangarajan, L. (2022). Recognition of compound characters in Kannada language.International Journal of Electrical and Computer Engineering(IJECE), 12(6), 6103-6113.

[5] Shelke, S., Apte, S. (2011). A Multistage Handwritten Marathi Compound Character Recognition Scheme using Neural Networks. International Journal of Signal Processing, Image Processing and Pattern Recognition, 4(1), 81-94

[6] Sachdeva, J., Mittal, S. (2021). Handwritten Offline Devanagari Compound Character Recognition Using Machine Learning.ACI'21: Workshop on Advances in Computational Intelligence at ISIC 2021, New Delhi, India.

[7] Kadam, A. A., Bhalerao M. V., (2019). Handwritten Marathi Compound Character Recognition.International Journal of Engineering Research & Technology (IJERT), 8(7), 742-747.

[8] Islam, M.S, Rahman, M.M, Rahman, M.H., Rivolta, M.W., Aktaruzzaman, M. (2022).RATNet: A Deep Learning Model for Bengali Handwritten Characters Recognition. Multimedia Tools and Applications, 81(8), 10631-10651.

[9] Sayeed,A., Shin, J., Hasan, A.L., Srizon, A.Y., Hasan, M.M. (2021). BengaliNet: A Low-Cost Novel Convolutional Neural Network for Bengali Handwritten Characters Recognition. Appl. Sci.,11(5),6845

[10] Pramanik, R., Bag, S. (2018). Shape decomposition-based handwritten compound character recognition for Bangla OCR. Journal of Visual Communication and Image Representation, 50,123–134.

[11] Das, N., Das, B., Sarkar, R., Basu, S., Kundu, M., &Nasipuri, M. (2010).Handwritten Bangla basic and compound character recognition using MLP and SVM classifier. arXiv preprint arXiv:1002.4040.

[12] Roy, S., Das, N., Kundu, M, Nasipuri, M. (2017). Handwritten Isolated Bangla Compound Character Recognition: a new benchmark using a novel deep learning approach. Pattern Recognition Letters, 90, 15-21.

[13] Pal, U., Wakabayashi, T., Kimura F. (2007).Handwritten Bangla Compound Character Recognition using Gradient Feature.10th International Conference on Information Technology, Orissa, 208- 213.

[14] Ashiquzzaman,A., Tushar, A.K., Dutta, S., Mohsin, F. (2017). An Efficient Method for Improving Classification Accuracy of Handwritten Bangla Compound Characters using DCNN with Dropout and ELU. Third International Conference on research in Compulligence and Communication Networks (ICRCICN), 147-152.

[15] Chakraborty, S., Paul, S. (2021). Bengali Handwritten Character Transformation: Basic to Compound and Compound to Basic Using Convolutional Neural Network. 2nd International Conference on Robotics, Electrical and Signal Processing Techniques (ICREST).
[16] Das, N., Acharya, K., Sarkar,R., Basu, S., Kundu, M.,Nasipuri, M. (2014). A benchmark image database of isolated Bangla handwritten compound characters. International Journal on Document Analysis and Recognition (IJDAR), 17(4),413-431.
[17] Garain U., Chaudhuri B. B. (1998). Compound Character Recognition By Run Number Based Metric Distance, Proc. of SPIE Annual Symposium on Electrical Imaging, , San Jose, California, USA, 98,Vol. 3305, 90-97.
[18] Muppalaneni, N.B. (2020). Handwritten Telugu Compound Character Prediction using Convolutional Neural Network.International Conference on Emerging Trends in Information Technology and Engineering (ic-ETITE).1-4.
[19] Sharif, S. M. A., Mohammed, N., Momen, S.,Mansoor, N (2018). Classification of Bangla Compound Characters Using a HOG-CNN Hybrid Model.Proceedings of the International Conference on Computing and Communication Systems, 403-411.
[20] Pradeep, J., Srinivasan, E. and Himavathi, S. (2011). Diagonal based feature extraction for handwritten Alphabets recognition system using neural network. International Journal of Computer Science and Information Technology, Vol. 3(1), 27-38.
[21] Chacko, B. P. and Anto, B. P. (2010).Pre and Post processing approaches in edge detection for character recognition. 12th International Conference on the Frontiers of Handwriting Recognition (ICFHR), 676-681
[22] Kumar, M., Jindal, M., Sharma, R., & Jindal, S. R. (2018). Offline handwritten numeral recognition using a combination of different feature extraction techniques.National Academy Science Letters, 41(1), 29–33.
[23] Arora, S., Bhattacharjee, D., Nasipuri, M., Basu, K., D. and Kundu, M. (2008).Combining multiple feature extraction techniques for handwritten Devnagari character recognition.10th Colloquium and 3rd International Conference on Industrial and Information Systems (ICIIS), 1-6.