DEEP HOG: A HYBRID MODEL TO CLASSIFY BANGLA ISOLATED ALPHA-NUMERICAL SYMBOLS

S.M.A. Sharif[*], *M. Mahboob*[†]

Abstract: Bangla is known to be the second most widely used script in the South Asian region. Despite its wide usage, a complete study with all available Bangla handwritten image classes is still due. This work proposes a hybrid model to classify all available handwritten image classes and unifying the existing benchmark datasets. The feasibility of the different handcrafted features in the hybrid model also has been demonstrated. Moreover, the proposed hybrid model obtain a maximum accuracy of 89.91 % in validation phase with a total of 259 Bangla alpha-numerical image classes. With the same number of image classes, the proposed hybrid model shows a testing accuracy of 89.28 % on 15,175 testing samples. The comparison results demonstrate that the proposed hybrid-HOG model can outperform the existing state-of-the-art classification models in Bangla handwritten alpha-numerical image classification. The code will be available on https://github.com/sharif-apu/hybrid-259.

Key words: *deep learning, hybrid-HOG, neural network, Bangla handwriting classification*

Received: September 9, 2017 **DOI:** 10.14311/NNW.2019.29.009
Revised and accepted: May 13, 2019

Introduction

Handwriting classification is apprised as one of the most significant research topics in the machine learning domain. It has gained the attention of many researchers due to the several real-life applications like number plate recognition, address identification, bank check recognition and so on [33]. As a consequence, different researchers already approached a different classification method to introduce an accurate and effective classifier. However, a robust classification system, which can handle a massive number of handwritten image classes for a widely used language like Bangla is still due.

Bangla is known as the second most widely used language script in the South Asian region. It has a rich collection of isolated character symbols. In general, there

[*]S.M.A. Sharif – Corresponding author; Department of Computer Science and Engineering, University of Liberal Arts, Bangladesh (ULAB), House No. 56, Road No. 4/A, Satmasjid Road, Dhaka 1209, Bangladesh, E-mail: sma.sharif.cse@ulab.edu.bd

[†]Mahdin Mahboob; Department of Electrical and Computer Engineering, Stony Brook University, 100 Nicolls Rd, Stony Brook, NY 11794, E-mail: mahdin.mahboob@stonybrook.edu

are 10 numerals and 50 isolated character symbols, which includes 11 vowel and 39 consonant characters. Like other Indian scripts (e.g., Devanagari [40]), Bangla also incorporated with a special character symbol set known as the compound characters. There around 334 compound characters with 171 unique pattern shapes can be found in the Bangla handwritten scripts [15]. Though these compound characters do not appear commonly as the regular characters, apart from that the compound characters have a significant impact in Bangla literature. However, the total number of pattern classes including multiple shapes of compound characters is around 259. Classifying a symbol set with a huge number of image classes is challenging. Furthermore, the multiple shapes of compound characters make handwritten Bangla character classification more complicated.

In the past decade, many classification approaches have been adopted in order to classify Bangla isolated character. In general, these classification approaches can be clustered into three basic categories according to their specific methods. In the early days, handcrafted feature extraction method like Local Binary Patterns(LBP) [1], Scale-invariant feature transform(SIFT) [25], Speeded-Up Robust Features (SURF) [3], Histogram of oriented gradients (HOG) [13] are apprised as the state of the art for the Bangla character classification. Typically, these features are extracted locally, globally or sometimes simultaneously, which again used to train a shallow classifier like Support Vector Machine (SVM), K-Means and so on. Later after the introduction of deep learning, the limelight from the shallow classifier and handcrafted features has been spotted out to the deep learning. Apart from that, two most recent studies [38, 39] introduced another hybrid classification approach for Bangla isolated character classification.

Despite the several novel classification approaches, Bangla character classification is far behind from the other languages. Unfortunately, all available works on Bangla character classification mostly emphasize specific symbol datasets rather than introducing a robust classifier, which can classify all available handwritten pattern classes simultaneously. In most recent works also focused on either numeral data sets or in character data sets. However, the lacking of a complete study on Bangla character classification is occurred due to two major reasons. First of all Bangla scripts contain more than 200 alpha-numeric symbols including special character. Secondly, the limitation with the existing benchmark datasets. Unfortunately, a language of over 250 million peoples only has a few numbers of open source data sets including a limited number of image classes. In this work, a complete study on isolated Bangla alpha-numerical has been demonstrated. Moreover, the feasibility of the existing deep network architecture to classify massive image classes also been revealed through the comparison experiments.

This work firstly focuses to make a unified data set from existing datasets. Thus it can cover all 259 image classes. Secondly, proposing a hybrid model, which can classify the unified data set. In general, the hybrid model [38, 39] is a combination of Convolutional Neural Network (CNN) and classical Artificial Neural Network (ANN). Thus, it can automatically extract features like CNN and also can learn handcrafted features by existing feature extraction methods. Thirdly, study the feasibility of different handcrafted features (e.g., SIFT [25], SURF [3], Gabor feature extraction method [32] and so on) in the hybrid model. Finally, a comparison between the proposed hybrid model and state-of-the-art classification

model (e.g., VGG16 [42], DenseNet [23], ResNet and so on) has been demonstrated. The overall work has been organized as follows: Section 1 highlights the existing work, Section 2 describes the datasets, proposed model. Results and comparison between different model have been analyzed in Section III. Section IV concludes the work.

1. Existing work

In the past decade, many researchers have put efforts in isolated character classification. Popular Indian scripts like Devanagari, Tamil and Bangla also draw special attention among all available Indian scripts. However, Bangla is considered as the most popular scripts after Devanagari. Despite its popularity, benchmark data set for Bangla character recognition is limited, especially related to the complete Bangla character set. In the '90s, B.B. Chaudhuri and U. Pal did work with Bangla characters classification. [11, 10, 34] Following their trends, many have come up with their innovative ideas to contribute to Bangla character classification.

Due to the simpler complexity, there are several works can be found with the Bangla numeral datasets [31, 16, 43]. In a study on a benchmark Bangla numeral data set, [4] proposed a multi-resolution wavelet analysis and majority voting approach in a multi-layer perceptron (MLP) based numeral networks to recognize the Bangla numeral classes. On their study, they have used 5000 training samples, which again validated with 1000 more samples from the same data set. In another study with the same numeral data set, [20] used LBP as their feature extraction method. Besides the shallow classifier, an adaptation of CNN or deep learning methods for Bangla handwriting numeral recognition also has been observed in recent years. For i.e. [2] utilized a CNN model classifying the Indian Statistical Institute (ISI) numeral data set. In addition, they achieved better performances compared to the shallow learners. On their study, they proposed a fixed augmentation method to enlarged the training dataset by making pivoted renditions of the training images.

On the other hand, Bangla character sets have a large number of pattern classes and their shapes are also complex compared to the numeral. Despite the importance of Bangla isolated characters, the recognition of Bangla isolated character classification is far behind compared to numeral classification. Very few online and off-line Bangla character classification works are found in the recent studies [35, 11, 7]. However, the reason for the low amount of work on Bangla basic character recognition is that there is a very limited number of benchmark datasets as well as they required a complex model to handle to handle a large number of classes more precisely.

Comparing to the numerals and isolated basic characters, work on Bangla compound character are very limited. In a study of Bangla Compound character, [14] applied a Genetic Algorithm (GA) and SVM-based approach to classifying CMATERdb 3.1.3.1 data set of 171 pattern classes, where the number of samples per character class were not equal and vary between 125 and 474 samples. On their follow up study, they achieved a better performance by using a Quad-tree based approach [15].

Besides the typical shallow classifier and deep learning approaches for Bangla handwriting classification, [38] introduced another hybrid model in the very recent year. They merge the HOG feature with a CNN model. In that study, they demonstrate that their hybrid model can outperform the existing classifiers even the deep CNN model as well. In their follow up study, they explore their model with the compound character and again outperform the existing classifier models, which are available for classifying Bangla compound character [39]. Unfortunately, they also didn't use their model to classify all available Bangla character classes. In addition, their hybrid model mostly focused on specific handcrafted feature (e.g., HOG feature extraction method). Apparently, the feasibility of other useful handcrafted features (e.g., SIFT [25], SURF [3], Gabor feature extraction method [32]) in hybrid model is still unknown.

Despite the plenty of novel works with a satisfactory amount of classification accuracy, all existing works are incorporated with specific types of symbols only. Thus, a study on Bangla character classes with a unified dataset, which includes all image classes is still missing. In addition, none of these studies have explored the efficacy of CNN combining with handcrafted features for a complete Bangla handwritten character classification. In this work, a unified dataset has been prepared by combining existing benchmark data sets. In addition, by adopting the concept of [38], a hybrid network has been proposed for classifying prepared unified dataset. Apparently, the proposed network model combines the use of features learned and extracted by CNN, with the very popular histogram of oriented gradients (HOG) image feature for a complete Bangla alpha-numerical dataset. Unlike the previous studies [38, 39], the feasibility of the other handcrafted features also has been studied for the unified dataset. In addition, an extensive comparison between the proposed model and state-of-the-art models has been demonstrated.

2. Present work

The present work has been relies on two fundamental stages. The first stage aimed to prepare a unify the existing benchmark datasets. This stage also includes data preprocessing and data augmentation. In the second stage a hybrid model has been introduced to classify the prepared unified dataset. Justification of the proposed model has been demonstrated in Section 3.

2.1 Dataset preparation

Though Bangla is a rich language, there are very few benchmark datasets are available for Bangla isolated character classification. Among all published datasets on Bangla, CMATERdb [36] and ISI Datasets [5] are widely used. The ISI dataset was developed by Indian Statistical Institute, Kolkata [5] and CMATERdb was developed by Jadavpur University, Kolkata [36]. In addition, CMATERdb is an open source dataset and have been used in several recent studies [36]. Moreover, the Indian Statistical Institute's dataset is not publicly available to download as the CMATERdb. This study uses these benchmark datasets to make a unified and larger dataset containing training, testing, and validate data samples.

Tab. I shows an overview of Indian Statistical Institute (ISI) dataset and CMATERdb with the corresponding number of sample images.

Name of Dataset	Training	Testing	Validation
ISI Numeral	19392	3996	–
ISI Basic Character	18000	12863	7000
CMATERdb Numeral	6000	–	–
CMATERdb Basic Character	12000	3001	–
CMATERdb Compound Character	33404	8381	–

Tab. I *Overview of the existing Bangla handwritten image dataset.*

2.1.1 Indian Stasitical Institute (ISI) dataset

The Indian Statistical Institute (ISI) dataset was developed for different Indian scripts including Devanagari, Bangla, Orya [6]. There are two types of datasets available for ISI. One is developed for online usage, another one is developed for offline handwriting recognition. In this work, the offline version of ISI Bangla Dataset(s) have been used. In addition, ISI offline dataset does not come in a unified form. There is a different subdivision of ISI dataset. Numeral [9], Basic Character, Compound character are the sub datasets available from ISI. According to [5], ISI dataset is standard and collected from different handwritten documents, postal mails, job application forms. In addition, few samples were also collected by a specially designed form.

The Indian Statistical Institute sample images are stored in tiff format with black text on a white background. Fig. 1 shows sample images from the ISI dataset.

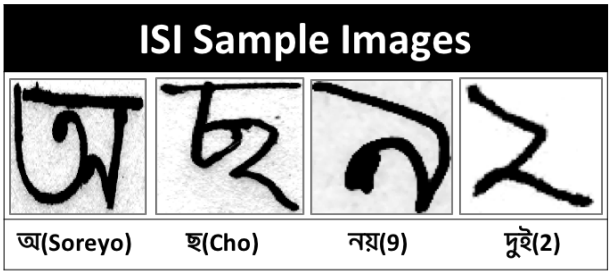

Fig. 1 *Image Samples from ISI datasets.*

2.1.2 CMATERdb

CMATERdb is an open source Bangla benchmark dataset. It also has a multi-language offline sample collection [36]. The CMATERdb 3.1.1, CMATERdb 3.1.3.1 are collections of handwritten Bangla numeral databases, isolated handwritten basic character and isolated Bangla compound-character databases. In addition, most

of the image in CMATERdb are stored in either as a grayscale image or binary images. Fig. 2 shows the sample images from CMATERdb databases.

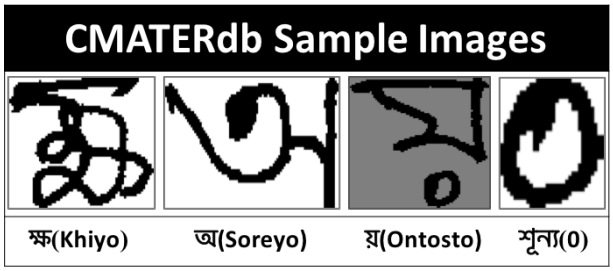

Fig. 2 *Image Samples from CMATERdb datasets.*

2.1.3 Unified dataset

Indian Statistical Institute datasets and CMATERdb was collected individually. The image quality and dimensions of data samples are also different. Moreover, the subdivision of each corresponding dataset contains a different number of sample images. In contrast, the proposed unified dataset has to have a common representation of the data samples along with 259 Bangla handwritten image classes. Thus, the sub-dataset of existing datasets has been merged into a single (unified) dataset for a general presentation. After that, the unified dataset has been divided into three sub-dataset as the convention of image classification studies followed in the past: the training dataset, testing dataset, and validation dataset. Tab. II demonstrates the overview of the unified dataset. At a glance, the unified dataset comprises of 82619 training, 25924 validation, and 15175 testing samples, which again obtained from the existing benchmark datasets.

Type	Training	Testing	Validation
Neumeral	19392	6000	3996
Basic Character	30000	13863	9000
Compound Character	33404	6181	2200
Total	82796	26044	15196
Actual	**82619**	**25924**	**15175**

Tab. II *Overview of unified dataset.*

2.1.4 Dataset preprocessing

Data preprocessing is one of the most important parts of this work. Due to the different image dimensions, all images of the unified dataset are resized into 28×28. Moreover, the backgrounds of all images are inverted in such a manner that all text appears in the white foreground in black background. Apart from that, the sample

images have been dilated by a morphological dilation filter [41]. Thus, the text information can be enhanced. Fig. 3 shows the steps of data preprocessing.

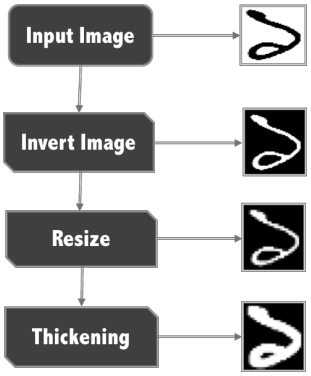

Fig. 3 *Data preprocessing steps.*

2.1.5 Data augmentation

Data augmentation is a method applicable to produce the multiple shallow or deep representation [8] of a source image. In machine learning, it is considered as an important strategy to enlarge the number of data samples [28, 45]. The main objective of data augmentation is to modify an image (I) by transformations (e.g., cropping and flipping, rotation and so on). In contrast, the basic information of the source image remains unchanged. In this work, three types of basic data augmentations have been applied. The first augmentation method is a random rotation of the training sample, where source image (I) has been randomly rotated between $(-50, +50)$ degree. Another augmentation has been done by applying block effect, thus the source image can lose some image quality [37]. The third augmentation considered as a wrapping operation, where the location information has been shifted between $(-5, +5)$ pixels horizontally or $(-5, +5)$ pixels vertically [37]. As a result, three versions of each training image has been obtained by the augmentation. Subsequently, the total number of training images is increased to 330476 training samples. Fig 4 shows sample images after data augmentation.

2.2 Model and experiment setup

In the recent past, Convolutional Neural Networks have been used as a base classifier because of their self-adaptive features [12] learning ability. Typically, a CNN takes the raw image as an Input for the particular classifier model. On the other hand, Hybrid models require an additional feature to learn from the feature vector alongside the raw images [38]. The abstract idea of the hybrid model is allowing the classifier model to learn handcrafted feature as well as a raw image through two different components. Typically, the output of the CNN models is considered as a 2D vector, where the output of a shallow classifier (used to train with handcrafted features) is always a 1D vector. Hence, the hybrid model always flattened

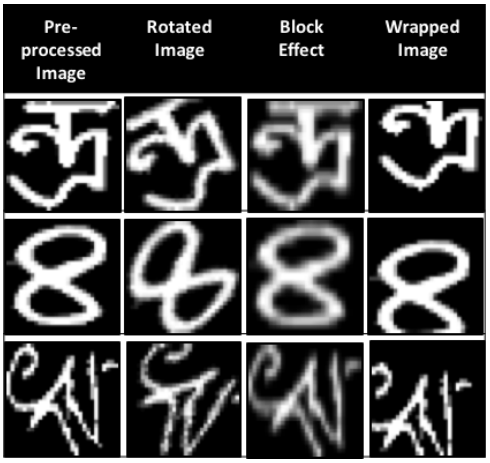

Fig. 4 *Augmented image samples.*

the feature vector of the CNN component into a 1D vector and merged it with the output of typical Artificial Neural Network (ANN). Finally, the concatenated feature vectors are pushed into a fully connected layer(s) for the final classifications. Fig. 5 shows the abstract idea of the hybrid model.

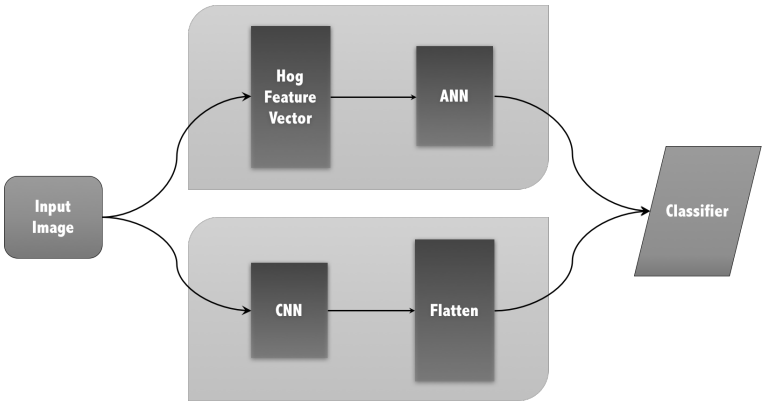

Fig. 5 *Concept of hybrid model.*

2.2.1 Architecture of proposed model

In this work, a hybrid model is proposed to classify the all Bangla alpha-numerical symbols. By following the convention of a hybrid model, the proposed model has three different components: CNN component, ANN with HOG and merged component. Fig. 6 demonstrates the model architecture of the proposed model. In general, the proposed model takes the gray scale images of 28×28 pixels as the input alongside the 72 dimension HOG feature vectors.

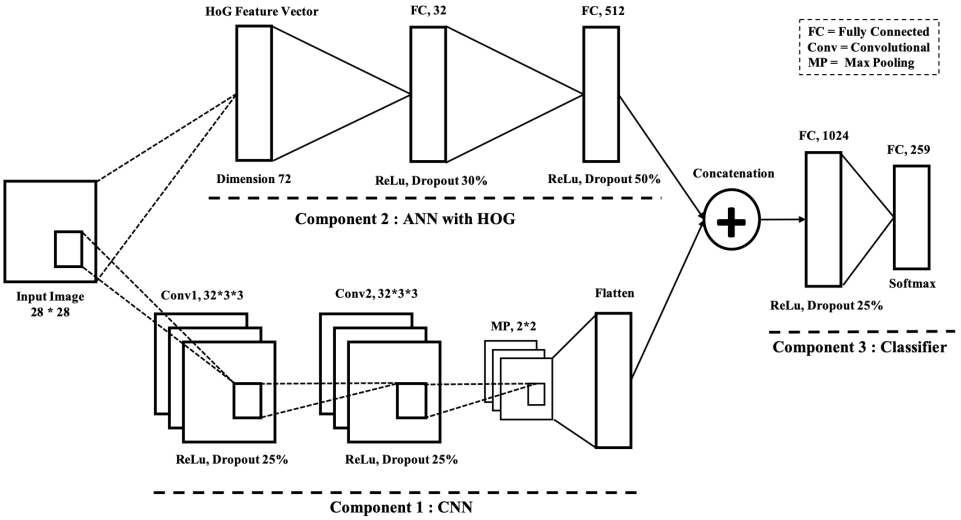

Fig. 6 *Architecture of the proposed model.*

- CNN Component: The first layer of CNN component comprise of a 32 convolutional filters with the kernel size of 3×3. The first layer has been followed by another convolutional layer with the same configuration. Moreover, the convolutional layers have been activated with a ReLu [30] function and overfitting has been controlled with a random dropouts (e.g., 25 % of dropout has been used) [18]. A max-pooling layer of 2×2 size [24] also been utilized to reduce the extracted features after the convolutional layers. Finally, the outcome of CNN component has been flattened into a 1D vector to concatenate with the output of the second component.

- HOG on ANN: The second component requires a 1D feature vector to feed the fully connected layers. Typically, human-engineered feature extraction methods are used to extract a feature from the corresponding input images. Prior to the current fame of deep learning, handcrafted feature (e. g., HOG and LBP) were considered as the state of the art, which was used to train a linear classifier e.g., SVM. In a prior work, [13] emphasized HOG as a convenient feature extraction method particularly for its simplicity and effectiveness at distinguishing shapes. Like previous studies [38, 39], this study also utilizes the HOG features to feed the ANN component. However, the feasibility of the other handcrafted features also demonstrated in the comparison experiments (please see section 3.1 for the details). In order to extract HOG features, typically the sample images has been divided into small locales which are called "cells". Each of the subdivided cells comprises of a cell size of 8 pixels. Then, the gradient is calculated in order to calculate the histogram of oriented gradients for each cell. Histogram bin size is set to 8 for each cell, which is concatenated into uniform histogram later. Here, the HOG feature vector has a dimensions of 72. Fig. 7 shows the HOG fea-

ture images extracted from the training data. The extracted HOG futures vectors has been fed into two consecutive hidden layers. The hidden layers comprise of 32 and 512 fully-connected nodes respectively. In addition, the fully-connected layers are activated with the ReLu functions [30] and followed by the dropout layers (30 % and 50 % dropout has been applied).

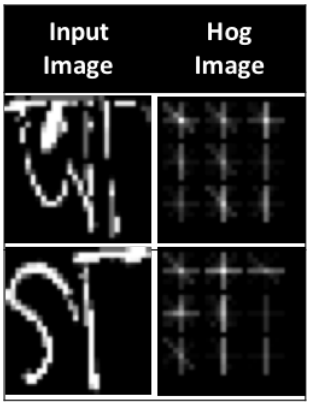

Fig. 7 *HOG image samples.*

- Classifier: The concatenated feature vector from the independent components (Component 1 and Component 2) have been used to feed the third components of the proposed model (here, this component performed as a softmax classifier [19]). In addition, the component 3 comprise of a hidden layer with the dimension of 1,024 and activated with the ReLu function. Moreover, a random dropout of 25 % has been introduced in order to stop the overfitting. The final layer of component 3 has a fixed dimensions of 259 (same number as the images classes) and activated with softmax activation fuction [17]. As a part of the experiments, three components has been treated as a single deep network and trained simultaneously.

The different variant of the proposed model has been studied through this work. Apparently, these variant are based on the individual component of the proposed model or they have been tweaked by parameters (e.g., depth, dropout, pooloing and so on) and features vectors (Please see section. 3 for more details).

2.2.2 Hyperperameters and model training

Every variant of the proposed model has been trained on the augmented training set of 330,476 images for 50 epochs(iterations). The learning rate of 0.001 has been used to train all models. In addition, the proposed model has been trained for 50 epochs (iteration). During the training phase, the best weight has been kept for further testing. To optimize the categorical cross entropy loss values, the Adadelta [44] optimizer was used in this work. All model training was done on two separate machines running on Ubuntu 16.04. In addition, the used machines are

equipped with two different GPU (NVIDIA GTX-670 and Titan XP) to accelerate the training process.

3. Result and experiments

To find the best combination of CNN and handcrafted features, several handcrafted features has been combined and experimented with the proposed model. The best handcrafted feature has been used to train the different variants of the proposed model. Moreover, the individual component of the proposed model has been justified with individual experiments. Finally, the state-of-the-art classification model has compared with proposed network. These experiments also revealed the feasibility of the existing networks structures in Bangla alpha-numerical image classification.

3.1 Comparison between handcrafted feature

To study the usefulness of existing handcrafted features on hybrid model, several experiments has been conducted in this work. However, the proposed network architectures is remain unchanged. The handcrafted features are used to feed the ANN component of the proposed model and overall network has been trained for 50 epochs. Tab. III shows the handcrafted features, brief about their extraction method, overall trainable parameters of the network, and the obtained results. Note that the used handcrafted features can have multiple variant and also can be tweaked by their required parameters. This study only focused on that variant which has been suggested by the recent works. However, the previous study on hybrid model [38, 39] reported that HOG feature can accelerate the network performance. Thus, the different parameters settings for the HOG features also been explored through the experiments.

As the Tab. III shows, HOG-8 outperformed the other handcrafted features in hybrid manner. Thus, in rest of the comparison experiments, HOG-8 has been used as handcrafted features. For the simplicity, HOG-8 has been denoted as simply HOG in the rest of the paper. In addition, rest of the experiments has been conducted with HOG features only.

3.2 Experiments with network variants

Including the proposed model, a total of 8 network variants have been shown in this work. Mostly these networks are the independent component of the proposed model. In addition, CNN component with a different number of parameters and different dropout rates have been experimented in this work. Though the dropout rate does not reduce the total number of parameters for a certain model, it helps to stop over-training for a particular network structure. Furthermore, to reduce the number of parameters and the feasibility of an additional max-pooling layer is also been studied. The best weights of the network variant have been used for the testing.

Feature	Extraction Method	Parameters	Val Acc. (%)
LBP [1]	3×3 kernel with radius of 8. Resultant a feature vector of dimension 128 [20]	5148291	88.59
SIFT [25]	128D SIFT with 4×4 Keypoints [26]	5258883	87.45
SURF [3]	64D SURF with 4×4 kernel and threshold 0.65 [26]	5146243	78.45
Gabor [32]	A local Gabor texture has been extracted with frequency .06. Feature dimension,of 28×28 (flatten into 786)	5169283	86.34
All Pixel (AP)	Gray scale (28×28) image has been flatten into 786	5169283	87.00
HOG-4 [13]	Cell Size of 4, bin size of 8, and the feature dimensions of 392	5156739	86.27
HOG-8 [13]	Cell Size of 8, bin size of 8, and the feature dimensions of 72	5146499	**89.91**
HOG-12 [13]	Cell Size of 12, bin size of 8, and the feature dimensions of 32	5145219	83.98
HOG-16 [13]	Cell Size of 16, bin size of 8, and the feature dimensions of 8	5144451	85.55

Tab. III *Handcrafted features in proposed model.*

3.2.1 Validation on network variants

Tab. IV shows the maximum validation accuracy and minimum loss observed in each model variants while conducting the experiments.

As Tab. IV shows, the HOG feature with CNN structure outperformed the CNN variants of the proposed model. In addition, a different number of filter and dropout rates can have an impact on the CNN variants. In CNN 1024_25 and CNN_50 models, the depth of the dense layer has been modified. In addition, these models also experiment with the different dropout rates (here, 25 and 50 represent the dropout rates). For these models, a small dropout rate can help the network to learn more useful information for such a large number of image classes. As the CNN models introduce a large number of trainable parameters, an additional max-pooling layer is also introduced after the first convolution in Max models. In contrast, reduction of trainable parameters also has a negative impact on the validation and training accuracy. From the experimental results, it is also visible that HOG features can accelerate model performance without introducing a massive number of parameters. Apparently, it also help the models to optimize the loss calculation. In sum up, the best result in the unified dataset is achieved by the Hybrid HOG 1024 model. It achieved the maximum validation accuracy of 89.91 % in its 20th iteration.

Model	Validation Accuracy (%)	Epoch	Training Loss (%)	Epoch
HOG ANN	44.45	13	3.55	49
CNN 512	85.96	30	0.04	48
CNN Max 512	88.58	39	0.35	43
HOG Max 512	**89.75**	**35**	**0.53**	**50**
CNN 1024_25	89.45	12	0.35	43
CNN 1024_50	88.07	26	0.95	50
Hybrid HOG 1024	**89.91**	**20**	**0.32**	**44**
HOG Max 1024	89.46	30	0.60	49

Tab. IV *Validation results for each experiments.*

Fig. 8 and Fig. 9 shows how the accuracy on the validation set varied during training and validation. Although only the best performing models were saved, it is evident from the comparison that the hybrid models consistently outperform the other models used during experiments.

Fig. 9 shows that CNN 512 model learns features faster than the other models. Apart from that, the lacking of dropouts has a negative impact on the validation (please see Fig. 8). However, the HOG features and dropout rates allows the proposed model to obtain minimum training loss. Fig. 10 shows the calculated loss values during training. In addition, it also shows the Hybrid model's consistency over each iteration.

Tab. VIII shows the indivisible results for each image classes and the Tab. V

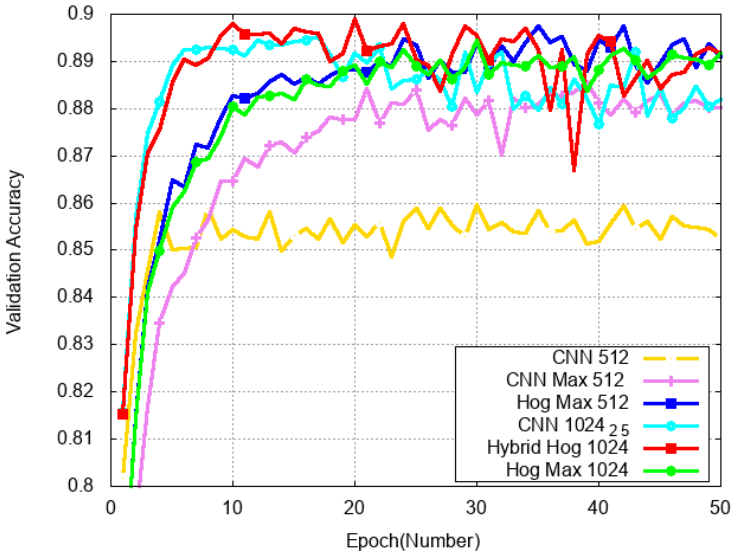

Fig. 8 *Validation accuracy obtained during validation phase.*

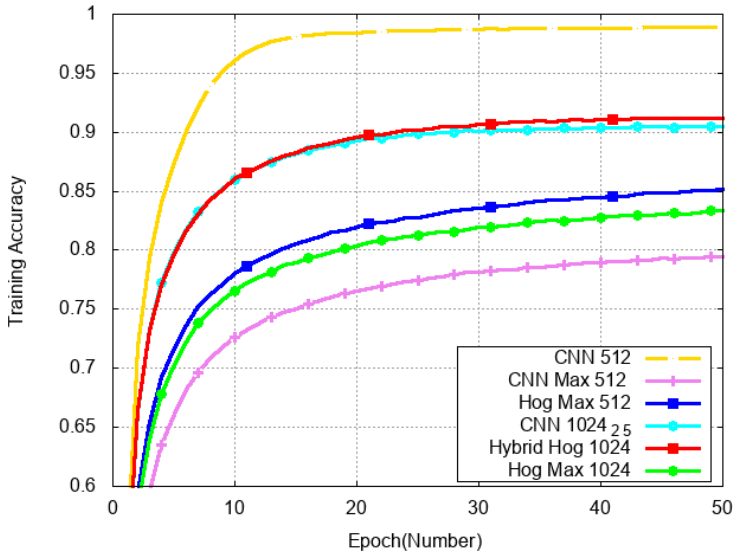

Fig. 9 *Training accuracy obtained over the training period.*

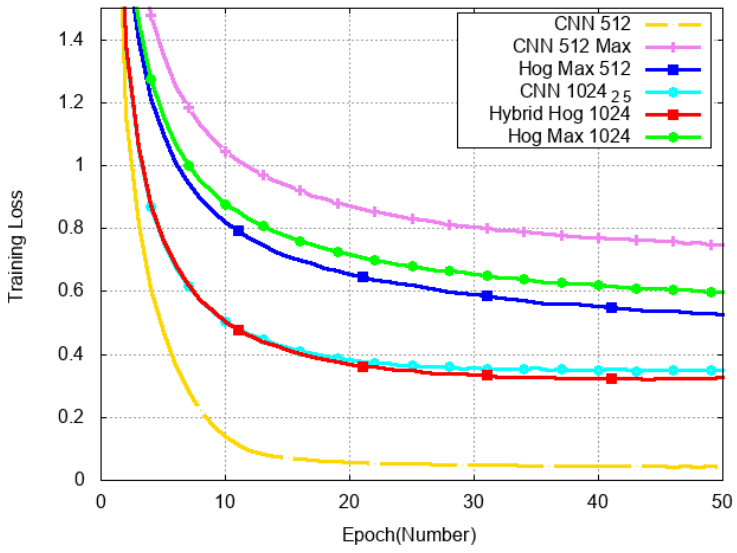

Fig. 10 *Training loss over the training period.*

illustrates the average of each symbol groups from the best weights of Hybrid HOG 1024. Furthermore, the Class Number 0-9 denotes the numerals, 10-59 are the isolated basic characters and rest of the image classes are either compound character or their multiple pattern shapes.

Symbol Type	Class Range	Average Accuracy (%)
Numeral	0-9	94.79
Basic Character	10-59	89.34
Compound Character	60-258	83.33

Tab. V *Average result obtained by each symbol types.*

Model	Testing Accuracy (%)	Epoch
CNN Max 512	88.47	39
CNN 1024_25	88.45	12
HOG Max 1024	88.74	30
HOG Max 512	89.05	35
Hybrid HOG 1024	89.28	20

Tab. VI *Testing on best weight.*

Tab. V show that numerals demonstrate the maximum average accuracy among all image groups. In opposite, compound characters demonstrate the minimum average accuracy because of their complex pattern shapes and less number of sample images.

3.2.2 Testing on network variants

Best preserved weights among all variants that demonstrate the validation accuracy above 89 % is tested on 25,924 image samples from the testing set of the unified dataset. Tab. VI shows the testing result on the best weights.

Network	Input Dim.	Paramters	Acc.(%)	Overfitting
LeNet (CNN) [29]	28×28	1498371	89.47	No
Autoencoder [27]	28×28	2862468	83.80	No
VGG16 [42]	$32 \times 32 \times 3$	14847555	89.14	Yes
VGG19 [42]	$32 \times 32 \times 3$	20157251	89.58	Yes
ResNet [21]	$32 \times 32 \times 3$	24118403	83.36	Yes
DenseNet [23]	$32 \times 32 \times 3$	7302979	86.55	Yes
MobileNet [22]	$32 \times 32 \times 3$	3494339	84.91	Yes
Proposed Model	28×28 and 72 (HOG feature)	5538179	**89.91**	**No**

Tab. VII *Comparison with state-of-the-art networks.*

Class	Sample	Identified	Acc(%)	Class	Sample	Identified	Acc(%)	Class	Sample	Identified	Acc(%)	Class	Sample	Identified	Acc(%)
0	400	400	100	65	11	9	81.82	130	11	11	100	195	11	11	100
1	400	375	93.75	66	11	9	81.82	131	11	6	54.55	196	11	10	90.91
2	400	382	95.5	67	11	10	90.91	132	11	10	90.91	197	11	11	100
3	397	361	90.93	68	11	10	90.91	133	11	11	100	198	11	8	72.73
4	400	399	99.75	69	11	11	100	134	11	5	45.45	199	11	11	100
5	396	349	88.13	70	11	11	100	135	11	10	90.91	200	11	11	100
6	400	389	97.25	71	11	11	100	136	11	3	27.27	201	11	10	90.91
7	400	396	99	72	11	8	72.73	137	11	11	100	202	11	11	100
8	398	397	99.75	73	11	10	90.91	138	11	10	90.91	203	11	11	100
9	395	331	83.8	74	11	9	81.82	139	11	11	100	204	11	9	81.82
10	180	172	95.56	75	11	11	100	140	11	10	90.91	205	11	11	100
11	180	172	95.56	76	11	10	90.91	141	11	6	54.55	206	11	8	72.73
12	180	167	92.78	77	11	9	81.82	142	11	6	54.55	207	11	8	72.73
13	180	158	87.78	78	11	10	90.91	143	11	9	81.82	208	11	8	72.73
14	180	165	91.67	79	11	10	90.91	144	11	11	100	209	11	3	27.27
15	180	158	87.78	80	11	8	72.73	145	11	9	81.82	210	11	9	81.82
16	180	150	83.33	81	11	10	90.91	146	11	8	72.73	211	11	11	100
17	180	150	83.33	82	11	6	54.55	147	11	11	100	212	11	3	27.27
18	180	159	88.33	83	11	9	81.82	148	11	11	100	213	11	5	45.45
19	180	144	80	84	11	10	90.91	149	11	7	63.64	214	11	9	81.82
20	180	140	77.78	85	11	11	100	150	11	11	100	215	11	9	81.82

Tab. VIII *Validation accuracy of each alpha-numerical symbols on the best weight of proposed model (part 1).*

Class	Sample	Identified	Acc(%)	Class	Sample	Identified	Acc(%)	Class	Sample	Identified	Acc(%)	Class	Sample	Identified	Acc(%)
21	180	162	90	86	11	10	90.91	151	11	7	63.64	216	11	7	63.64
22	180	144	80	87	11	8	72.73	152	11	10	90.91	217	11	7	63.64
23	180	166	92.22	88	11	9	81.82	153	11	11	100	218	11	9	81.82
24	180	155	86.11	89	11	10	90.91	154	11	11	100	219	11	11	100
25	180	161	89.44	90	11	10	90.91	155	11	10	90.91	220	11	11	100
26	180	176	97.78	91	11	7	63.64	156	11	10	90.91	221	11	10	90.91
27	180	171	95	92	11	9	81.82	157	11	8	72.73	222	11	9	81.82
28	180	170	94.44	93	11	8	72.73	158	11	9	81.82	223	11	10	90.91
29	180	168	93.33	94	11	9	81.82	159	11	9	81.82	224	11	11	100
30	180	172	95.56	95	11	9	81.82	160	11	9	81.82	225	11	11	100
31	180	165	91.67	96	11	10	90.91	161	11	9	81.82	226	11	11	100
32	180	149	82.78	97	11	10	90.91	162	11	10	90.91	227	11	11	100
33	180	127	70.56	98	11	10	90.91	163	11	8	72.73	228	11	9	81.82
34	180	167	92.78	99	11	11	100	164	11	10	90.91	229	11	9	81.82
35	180	161	89.44	100	11	10	90.91	165	11	8	72.73	230	11	10	90.91
36	180	98	54.44	101	11	10	90.91	166	11	11	100	231	11	5	45.45
37	180	168	93.33	102	11	1	9.09	167	11	10	90.91	232	11	11	100
38	180	174	96.67	103	11	9	81.82	168	11	11	100	233	11	10	90.91
39	180	172	95.56	104	11	3	27.27	169	11	10	90.91	234	11	9	81.82
40	180	154	85.56	105	11	8	72.73	170	11	9	81.82	235	11	8	72.73
41	180	162	90	106	11	5	45.45	171	11	11	100	236	11	11	100
42	180	166	92.22	107	11	10	90.91	172	11	10	90.91	237	11	10	90.91

Tab. III *Validation accuracy of each alpha-numerical symbols on the best weight of proposed model (part 2).*

Class	Sample	Identified	Acc(%)	Class	Sample	Identified	Acc(%)	Class	Sample	Identified	Acc(%)	Class	Sample	Identified	Acc(%)
43	180	175	97.22	108	11	10	90.91	173	11	1	9.09	238	11	9	81.82
44	180	150	83.33	109	11	6	54.55	174	11	9	81.82	239	11	11	100
45	180	164	91.11	110	11	9	81.82	175	11	9	81.82	240	11	9	81.82
46	180	140	77.78	111	11	3	27.27	176	11	10	90.91	241	11	9	81.82
47	180	174	96.67	112	11	8	72.73	177	11	7	63.64	242	11	11	100
48	180	167	92.78	113	11	10	90.91	178	11	11	100	243	11	7	63.64
49	180	164	91.11	114	11	10	90.91	179	11	6	54.55	244	11	11	100
50	180	166	92.22	115	11	10	90.91	180	11	10	90.91	245	11	9	81.82
51	180	155	86.11	116	11	10	90.91	181	11	9	81.82	246	11	10	90.91
52	180	152	84.44	117	11	8	72.73	182	11	11	100	247	11	9	81.82
53	180	173	96.11	118	11	11	100	183	11	11	100	248	11	11	100
54	180	175	97.22	119	11	10	90.91	184	11	10	90.91	249	11	10	90.91
55	180	173	96.11	120	11	11	100	185	11	9	81.82	250	11	11	100
56	180	167	92.78	121	11	9	81.82	186	11	10	90.91	251	11	11	100
57	180	171	95	122	11	11	100	187	11	9	81.82	252	11	9	81.82
58	180	166	92.22	123	11	10	90.91	188	11	10	90.91	253	11	9	81.82
59	180	166	92.22	124	11	11	100	189	11	10	90.91	254	11	0	0
60	11	7	63.64	125	11	11	100	190	11	8	72.73	255	11	11	100
61	11	10	90.91	126	11	10	90.91	191	11	9	81.82	256	11	11	100
62	11	10	90.91	127	11	7	63.64	192	11	10	90.91	257	11	10	90.91
63	11	11	100	128	11	9	81.82	193	11	10	90.91	258	11	10	90.91
64	11	9	81.82	129	11	8	72.73	194	11	5	45.45	–	–	–	–

Tab. III *Validation accuracy of each alpha-numerical symbols on the best weight of proposed model (part 3).*

In the testing phase, the proposed Hybrid Model also outperformed the other models and shows consistency. This model is able to learn features from large and complex datasets better than the other models varients compared in this work.

3.3 Comparison with state-of-the-art networks

The state of the art Deep Network has been studied to find their feasibility in Bangla alphanumeric symbol classification. In order to do so, the existing deep model [29, 27, 42, 21, 23, 22] has been trained with their suggested hyperperamarters. Note that the none of these networks are designed to classify the Bangla alphanumerical symbol with 259 image classes. As a consequences., the output layer of the each model has been modified to predict the 259 image classes. Thus, the actual parameters of that model could be different from the number demonstrate in the Tab. VII. For the fair comparison, all of the used deep network has been trained with the augmented unified dataset. A monitoring log has been maintained to observed the overfitting while training each models. Initially it has been aimed to train each model for 50 iteration. However, considering the effect of over fitting a few network has been stopped earlier. Tab. VII shows the performance of existing deep networks on unified dataset.

As the Tab. VII shows, the proposed hybrid model outperform existing state-of-the-art classification model for classifying Bangla handwritten alpha-numerical image classes. Unlike other existing networks with deeper architecture (e.g., VGG, ResNet, DenseNet) it does not suffer from overfitting during the validation phase.

4. Conclusion

A method for unifying existing Bangla handwritten character dataset has been proposed in this study, where the unified dataset contains 259 Bangla handwritten image classes. Moreover, a hybrid deep model has been proposed, which combines a convolutional neural network with HOG features. The impact of the different handcrafted features also been demonstrated in this study. The proposed model achieves validation accuracy of 89.91 % in the augmented version of the unified dataset. Alongside the validation, the testing result also demonstrates the consistency of the proposed model. The comparison experiments show that the proposed model can outperform the existing state-of-the-art classification models for classifying Bangla alpha-numerical symbols. It has been planned to study the feasibility of the hybrid model for the more complex dataset. All scripts, including preparing dataset from the existing dataset, best weight, model training scripts, etc. are uploaded into GitHub and open sourced to others for further development.

Acknowledgement

Datasets used in this study were collected from the Indian Statistical Institute and Jadavpur University, Kolkata. This study is a self-motivated research work and not funded by any organization. The authors are thankful to Gachon Computer Vision and Image Processing (CVIP) lab, South Korea for allowing to use their hardware for the comparison experiments.

References

[1] AHONEN T., HADID A., PIETIKAINEN M. Face description with local binary patterns: Application to face recognition. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*. 2006, 28(12), pp. pp.2037–2041.

[2] AKHAND M., AHMED M., RAHMAN M.H. Convolutional neural network training with artificial pattern for Bangla handwritten numeral recognition. In: *Informatics, Electronics and Vision (ICIEV), 2016 5th International Conference on*, 2016, pp. 625–630. doi: 10.1109/ICIEV.2016.7760077,

[3] BAY H., TUYTELAARS T., VAN GOOL L. Surf: Speeded up robust features. In: *European conference on computer vision*, 2006, pp. 404–417. doi: 10.1.1.407.7433,

[4] BHATTACHARYA U., CHAUDHURI B. A majority voting scheme for multiresolution recognition of handprinted numerals. In: *Document Analysis and Recognition, 2003. Proceedings. Seventh International Conference on*, 2003, pp. 16–20. doi: 10.1109/ICDAR.2003.1227620,

[5] BHATTACHARYA U., CHAUDHURI B. Databases for research on recognition of handwritten characters of Indian scripts. In: *Document Analysis and Recognition, 2005. Proceedings. Eighth International Conference on*, 2005, pp. 789–793. doi: 10.1109/ICDAR.2005.84,

[6] BHATTACHARYA U., CHAUDHURI B.B. Handwritten numeral databases of Indian scripts and multistage recognition of mixed numerals. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*. 2009, 31(3), pp. pp.444–457.

[7] BHATTACHARYA U., SHRIDHAR M., PARUI S.K. On recognition of handwritten Bangla characters. In: *Computer Vision, Graphics and Image Processing*. 2006, pp. 817–828. doi: 10.1007/11949619_73.

[8] CHATFIELD K., SIMONYAN K., VEDALDI A., ZISSERMAN A. Return of the devil in the details: Delving deep into convolutional nets. *arXiv preprint arXiv:1405.3531*. 2014.

[9] CHAUDHURI B.B. A complete handwritten numeral database of Bangla - A major Indic script. In: *Proc. 10th IWFHR, La Baule*, 2006, pp. 379–384. doi: 10.1.1.496.6920,

[10] CHAUDHURI B., PAL U. A complete printed Bangla OCR system. *Pattern recognition*. 1998, 31(5), pp. pp.531–549.

[11] CHAUDHURI B., PAL U. An OCR system to read two Indian language scripts: Bangla and Devnagari (Hindi). In: *Document Analysis and Recognition, 1997., Proceedings of the Fourth International Conference on*, 1997, pp. 1011–1015. doi: 10.1109/ICDAR.1997.620662,

[12] CIRESAN D.C., MEIER U., GAMBARDELLA L.M., SCHMIDHUBER J. Convolutional neural network committees for handwritten character classification. In: *Document Analysis and Recognition (ICDAR), 2011 International Conference on*, 2011, pp. 1135–1139. doi: 10.1109/ICDAR.2011.229,

[13] DALAL N., TRIGGS B. Histograms of oriented gradients for human detection. In: *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, 2005, pp. 886–893. doi: 10.1109/CVPR.2005.177,

[14] DAS N., ACHARYA K., SARKAR R., BASU S., KUNDU M., NASIPURI M. A novel GA-SVM based multistage approach for recognition of handwritten Bangla compound characters. In: *Proceedings of the International Conference on Information Systems Design and Intelligent Applications 2012 (INDIA 2012) held in Visakhapatnam, India, January 2012*, 2012, pp. 145–152. doi: 10.1007/978-3-642-27443-5_17,

[15] DAS N., ACHARYA K., SARKAR R., BASU S., KUPP.NDU M., NASIPURI M. A benchmark image database of isolated Bangla handwritten compound characters. *International Journal on Document Analysis and Recognition (IJDAR)*. 2014, 17(4), pp. pp.413–431.

[16] DAS N., REDDY J.M., SARKAR R., BASU S., KUNDU M., NASIPURI M., BASU D.K. A statistical–topological feature combination for recognition of handwritten numerals. *Applied Soft Computing*. 2012, 12(8), pp. pp.2486–2495.

[17] DUNNE R.A., CAMPBELL N.A. On the pairing of the softmax activation and cross-entropy penalty functions and the derivation of the softmax activation function. In: *Proc. 8th Aust. Conf. on the Neural Networks, Melbourne*, 1997, pp. pp.185. doi: 10.1.1.49.6403,

[18] ELLEUCH M., MAALEJ R., KHERALLAH M. A new design based-SVM of the CNN classifier architecture with dropout for offline Arabic handwritten recognition. *Procedia Computer Science*. 2016, 80, pp. pp.1712–1723.

[19] GRAVE E., JOULIN A., CISSÉ M., JÉGOU H., et al. Efficient softmax approximation for GPUs. In: *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, 2017, pp. 1302–1310. doi: 10.1080/02564602.2015.1015631,

[20] HASSAN T., KHAN H.A. Handwritten Bangla numeral recognition using Local Binary Pattern. In: *Electrical Engineering and Information Communication Technology (ICEEICT), 2015 International Conference on*, 2015, pp. 1–4. doi: 10.1109/ICEEICT.2015.7307371,

[21] HE K., ZHANG X., REN S., SUN J. Deep residual learning for image recognition. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778. doi: 10.1109/CVPR.2016.90,

[22] HOWARD A.G., ZHU M., CHEN B., KALENICHENKO D., WANG W., WEYAND T., ANDREETTO M., ADAM H. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*. 2017.

[23] HUANG G., LIU Z., VAN DER MAATEN L., WEINBERGER K.Q. Densely connected convolutional networks. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 4700–4708. doi: 10.1109/CVPR.2017.243,

[24] JARRETT K., KAVUKCUOGLU K., LECUN Y., et al. What is the best multi-stage architecture for object recognition? In: *Computer Vision, 2009 IEEE 12th International Conference on*, 2009, pp. 2146–2153. doi: 10.1109/ICCV.2009.5459469,

[25] KE Y., SUKTHANKAR R. PCA-SIFT: A more distinctive representation for local image descriptors. In: *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*, 2004, pp. II–506. doi: 10.1109/CVPR.2004.1315206,

[26] KHAN N.Y., MCCANE B., WYVILL G. SIFT and SURF performance evaluation against various image deformations on benchmark dataset. In: *2011 International Conference on Digital Image Computing: Techniques and Applications*, 2011, pp. 501–506. doi: 10.1109/DICTA.2011.90,

[27] KINGMA D.P., WELLING M. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*. 2013.

[28] KRIZHEVSKY A., SUTSKEVER I., HINTON G.E. Imagenet classification with deep convolutional neural networks. In: *Advances in neural information processing systems*, 2012, pp. 1097–1105. doi: 10.1.1.299.205,

[29] LECUN Y., BENGIO Y. Convolutional networks for images, speech, and time series. *The handbook of brain theory and neural networks*. 1995, 3361(10), pp. pp.1995.

[30] LI Y., YUAN Y. Convergence analysis of two-layer neural networks with relu activation. In: *Advances in Neural Information Processing Systems*, 2017, pp. 597–607,

[31] LIU C.-L., SUEN C.Y. A new benchmark on the recognition of handwritten Bangla and Farsi numeral characters. *Pattern Recognition*. 2009, 42(12), pp. pp.3287–3295.

[32] LIU C., WECHSLER H. Gabor feature based classification using the enhanced fisher linear discriminant model for face recognition. *IEEE Transactions on Image processing*. 2002, 11(4), pp. pp.467–476.

[33] PAL U, CHAUDHURI B. Indian script character recognition: a survey. *pattern Recognition*. 2004, 37(9), pp. pp.1887–1899.

[34] PAL U, CHAUDHURI B. OCR in Bangla: an Indo-Bangladeshi language. In: *Pattern Recognition, 1994. Vol. 2-Conference B: Computer Vision & Image Processing., Proceedings of the 12th IAPR International. Conference on*, 1994, pp. 269–273. doi: 10.1109/ICPR.1994.576917,

[35] PARUI S.K., GUIN K., BHATTACHARYA U., CHAUDHURI B.B. Online handwritten Bangla character recognition using HMM. In: *Pattern Recognition, 2008. ICPR 2008. 19th International Conference on*, 2008, pp. 1–4. doi: 10.1109/ICPR.2008.4761835,

[36] SARKAR R., DAS N., BASU S., KUNDU M., NASIPURI M., BASU D.K. CMATERdb1: a database of unconstrained handwritten Bangla and Bangla–English mixed script document image. *International Journal on Document Analysis and Recognition (IJDAR)*. 2012, 15(1), pp. pp.71–83.

[37] SHARIF S., MAHBOOB M. Evil method: A deep CNN model for Bangla handwritten numeral classification. In: *2017 4th International Conference on Advances in Electrical Engineering (ICAEE)*, 2017, pp. 217–222. doi: 10.1109/ICAEE.2017.8255356,

[38] SHARIF S., MOHAMMED N., MANSOOR N., MOMEN S. A hybrid deep model with HOG features for Bangla handwritten numeral classification. In: *Electrical and Computer Engineering (ICECE), 2016 9th International Conference on*, 2016, pp. 463–466. doi: 10.1109/ICECE.2016.7853957,

[39] SHARIF S., MOHAMMED N., MOMEN S., MANSOOR N. Classification of Bangla Compound Characters Using a HOG-CNN Hybrid Model. In: *Proceedings of the International Conference on Computing and Communication Systems*, 2018, pp. 403–411. doi: 10.1007/978-981-10-6890-4_39,

[40] SHELKE S., APTE S. Multistage handwritten marathi compound character recognition using neural networks. *Journal of Pattern Recognition Research*. 2011, 2(pp.253-268).

[41] SIGMUND O. Morphology-based black and white filters for topology optimization. *Structural and Multidisciplinary Optimization*. 2007, 33(4-5), pp. pp.401–424.

[42] SIMONYAN K., ZISSERMAN A. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*. 2014.

[43] WEN Y., LU Y., SHI P. Handwritten Bangla numeral recognition system and its application to postal automation. *Pattern recognition*. 2007, 40(1), pp. pp.99–107.

[44] ZEILER M.D. ADADELTA: an adaptive learning rate method. *arXiv preprint arXiv:1212.5701*. 2012.

[45] ZEILER M.D., FERGUS R. Visualizing and understanding convolutional networks. In: *European Conference on Computer Vision*, 2014, pp. 818–833. doi: 10.1007/978-3-319-10590-1_53,