Devanagari Character Recognition in the Wild

O. V. Ramana Murthy
Dept of Electrical Engg
IIT Delhi, India

Sujoy Roy
Institute for Infocomm Research,
A-Star, Singapore

Vipin Narang
Singapore Immunology Network (SIgN)
A-Star, Singapore

M. Hanmandlu
Dept of Electrical Engg
IIT Delhi, India

ABSTRACT
This papers examines the issues in recognizing the Devanagari characters in the wild like sign boards, advertisements, logos, shop names, notices, address posts etc. While some works deal with the issues in recognizing the machine printed and the handwritten Devanagari characters, it is not clear if such techniques can be directly applied to the Devanagari characters captured in the wild. Moreover in the recent times a lot of research has been conducted in the field of object categorization and localization. It would be interesting to investigate if the state-of-the-art tools for object categorization can also be applied to the recognition of the Devanagari characters. The idea is to view the isolated characters as objects so as to detect them in the wild. The ability to recognize the Devanagari characters in the wild will be very useful in the Internet services like Google street view and its associated applications. So, a detailed study of the Devanagari character recognition using the state-of-the-art character recognition and object recognition tools has been carried out to compute the best performance. This serve as a baseline for the comparison for the future works.

There is no benchmark database to conduct studies on the Devanagari character recognition in the wild. So a database of 40 Devanagari character categories has been created from 200 pictures of the images in the wild.

GENERAL TERMS: Character recognition, Devanagari script, local and global feature selection, object recognition.

KEYWORDS: Object recognition, camera-based character recognition, Devanagari characters, off-line handwritten character recognition.

1. INTRODUCTION
Recognizing characters has been widely understood as a means of mechanizing the process of understanding text in the written form to facilitate fast and efficient use. However, text is all around us through '*text in the wild*' that encompasses street signs, shop names, product advertisements, posters on streets etc. In fact we actively interact with the world at large through such *text in the wild*. Think of a tourist in a far off land where he does not understand the local language and finds it hard to communicate with the local people in getting his things done. Living in a multicultural society, we are supposed to understand at least more than one language to function confidentially in our daily transactions. This fact stresses the need for automatically recognizing the text in the wild.

A survey on the state-of the-art methodologies on character recognition from the wild can be found in [8]. Most of the works on recognizing characters from the wild pertain to English characters only. de Campos *et al.* [5] have worked on the recognition of English and Kannada characters from the images captured from the streets of Bangalore. They have used the bag-of-visual-words based object categorization framework for the character recognition. The necessity for the custom based solutions is advocated by them in lieu of commercial OCR solutions. They conclude that the character recognition of typical Indian languages, like Kannada is extremely challenging due to the large number of visually distinct classes formed out of different combinations of the basic alphabets. However such study on Devanagari script is at the nascent stage.

The paper is organized as follows. Section 2 gives a description of the Devanagari script. Section 3 describes the creation of the databases. Section 4 describes the preprocessing and the various state-of-the-art feature representations used in this work. Section 5 contains the results obtained by application of these features and classifiers for the recognition of the Devanagari characters in the wild. Finally conclusions are drawn in section 6.

2. DESCRIPTION OF THE DEVANAGARI SCRIPT
Devanagari script is written from left to right and it does not have any upper or lower case letters. It is usually recognized by a horizontal line that connects the top of the characters in a word. However, in some words, all the characters are not connected. The alphabets consisting of consonants, vowels, conjuncts in the Devanagari script are now enumerated.

Consonants: There are 33 consonants (shown in Fig. 1(a)).

Vowels: The vowels (see Fig. 1(b)) in Devanagari, similar to English have two characteristic features.

a. Each vowel has a sound associated with it.
b. Vowels modify the sound of a consonant. In order to modify the sound of a consonant, a modifier symbol (shown in Fig. 1(c)) is attached to the consonant at the appropriate location. A vowel is represented in a modified shape known as "*matra*". To elucidate the use of *matra*, consider the first consonant of the script 'क' and a *matra*, shown in the second row of Fig. 1 (c).

Conjuncts: Sometimes two or more consonants are combined together to form a composite alphabet, called conjunct. An example of conjunct is shown in the Fig. 1 (d).

क	ख	ग	घ	ङ
च	छ	ज	झ	ञ
ट	ठ	ड	ढ	ण
त	थ	द	ध	न
प	फ	ब	भ	म
य	र	ल	व	श
ष	स	ह		

(a) Consonants

अ	आ	इ	ई	उ	ऊ
ऋ	ए	ऐ	ओ	औ	

(b) Vowels

Matra	ा	ि	ी	ु	ू
Modified Character	**का**	**कि**	**की**	**कु**	**कू**
Matra	ृ	े	ै	ो	ौ
Modified Character	**कृ**	**के**	**कै**	**को**	**कौ**

(c)*Matra* symbols corresponding to the vowels

	क	ख	ग
क	क्क	क्ख	क्ग
ख	ख्क	ख्ख	ख्ग
ग	ग्क	ग्ख	ग्ग

(d) Conjuncts

Fig. 1 Devanagari script

3. DATABASE

An important requirement to validate a character recognition method is to have a benchmark database covering all the varieties. Three databases employed are: DSIW-3K (Devanagari Script in the Wild), DSMP-8K, DSMP-48K (Devanagari Script in the machine printed form) and DSHnd-30K (Devanagari Script in the handwritten form). Their details are given here:

a) **DSIW-3K:** A database of the Devanagari characters is collected from pictures of signboards, hoardings and advertisements in streets, shopping areas, roadside signs, and public areas like parks etc. Text strings are manually extracted from these pictures and a heuristic segmentation technique is used to segment the text strings into individual characters. Note that segments can contain characters with *matras* or conjuncts.

The text images in the wild are taken outdoors in the unconstrained settings using a Nokia Mobile camera with a resolution of 1.5Megapixels. The background, texture, size and positioning of character fonts do not follow strict margins or settings as shown in Fig. 2(a). Pictures are taken under real-life situations without caring about the refined camera settings and environmental conditions, which cause poor resolution and distortion (see Fig. 2(b)).

The database contains an unequal number of samples for each conjuncts/characters as these are extracted from the text strings in the images. This is also in line with the fact that in daily usage some characters are more frequently used than others. The category and number of samples collected therein are shown in Table 1**.**

The characters are also categorized according to their semantics. For instance, कि is categorized into two classes- the consonant class क and matra class ि. Fig. 3 and Fig. 4 present examples of characters and *matras* from the real world image data set.

(a)

(b)

Fig. 2 Some of the characters in the wild (a) With unconstrained artistic fonts and background (b) Geometric distortions, noise due to specular reflection.

Fig. 3 A variety of samples for the category ट

b) **DSHnd-30K:** This dataset consists of 30,355 samples of the handwritten Devanagari characters. This is a subset of the

database provided by ISI [13, 14] created for the OCR of the scanned documents. These samples are collected from the designed user fill-in sheets. The number of samples for each character class is given in Table 2. Samples for classes 39 and 40 do not exist in the original database provided by Pal *et al* [13, 14]. So samples for क्ष and त्र are collected separately.

Fig. 4 A variety of samples for the category ु

Table 1 Details of the dataset DSIW-3K

Character	No. of samples	Character	No. of samples	Character	No. of samples	Character	No. of samples
अ	56	द	59	ढ	3	्	0
इ	12	ध	24	ण	1	ा	418
उ	7	न	161	त	3	ि	126
ऋ	0	य	78	थ	11	ी	188
ए	34	र	173	द	2	ु	66
क	182	ल	143	ध	0	ो	116
ख	22	व	96	न	1	ौ	11
ग	77	श	41	प	29	ं	65
घ	11	ष	13	ब	8	ँ	8
ङ	1	स	151	भ	10	०	2
च	39	ह	91	म	8	१	1
छ	14	क्ष	5	य	0	२	5
ज	99	ज्ञ	1	र	14	३	2
झ	1	प	111	ल	13	४	1
ञ	15	फ	27	ळ	2	५	0
ट	104	ब	72	श	5	६	2
ठ	6	भ	39	ष	7	७	1
ड	78	म	110	स	36	८	0
ढ	2	त	19	ह	31	९	0
ण	12	क	23	ृ	12	श्र	5
त	70	ऋ	3	ॅ	147		
थ	19	ग	4	ै	55		

Table 2 Details of the dataset DSHnd-30K

Class no.	Character	No. of samples	Class no.	Character	No. of samples
1		789	22		739
2		784	23		760
3		787	24		681
4		775	25		780
5		709	26		806
6		784	27		777
7		787	28		676
8		783	29		764
9		864	30		783
10		781	31		784
11		783	32		777
12		778	33		779
13		784	34		860
14		768	35		779
15		775	36		714
16		787	37		766
17		780	38		756
18		771	39		366
19		762	40		576
20		782			
21		781			

c) **DSMP-4K**: This dataset consists of 3840 samples of the machine printed Devanagari characters. The 40 classes, as shown in Table 2 are chosen for this study. The number of classes is same as in DSHnd-30K. The basic set of 3840 (40×24×4) samples is created from 24 different Devanagari fonts, in their Normal, Italic, Bold and Italic Bold styles.

d) **DSMP-24K:** This dataset is a transformation of the DSMP-4K. It may be noted that 6 kinds transformation are applied on the basic set to create the *transformed* set. This gives 23,040 samples of characters (3840×6 = 23040). A sample of the

transformed characters generated from a character 'अ' is shown in Fig. 5. The transformations applied include (a) Barrel distortion (due to lens) (b) pin cushion distortion (due to lens) (c) Projective transformation (d)-(e) random rotation in the range [10, 60] degrees and (f) shearing transformation. This dataset does not have any semantic categories.

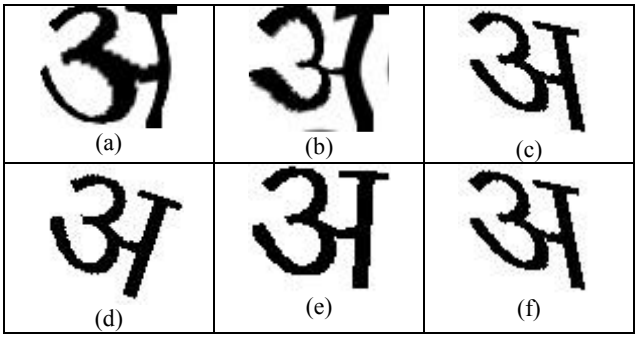

Fig. 5 Character 'अ' undergoing transformations (a) Barrel distortion (due to lens) (b) Pin cushion distortion due to lens (c) Projective transformation (d) Rotation of 60^o (e) Rotation of 15^o and (f) Shearing transformation.

We note that several characters seen in the wild are the transformations of the machine printed characters. Moreover to learn a model of characters, we need a collection of representative samples, which is very hard to obtain from the real world images. Hence the transformation of the machine printed characters is studied to learn the Devanagari characters in the wild. The efficacy of using the machine printed characters to train a model and then test on the images in the wild is investigated by de Campos *et al* [5] on the Kannada character recognition problem and by Neuman and Matas [10], Kai Wang *et al* [6] on the English Character recognition.

e) **DSMP-28K:** This data set is a combination of DSMP-4K and DSMP-24K.

4. PREPROCESSING AND FEATURES

4.1 Preprocessing

Preprocessing is applied to any character to modify the segmented character to a form best suitable for feature extraction. This modification usually involves the application of several image enhancement techniques. The following is the sequence of operations employed in the processing:

Step 1: Smoothing – The raw character is converted into a grayscale version and then smoothed by applying a 2 x 2 mean filtering 4 times.
Step 2: Binarization - The smoothed image is then converted into a binary image by using Otsu's threshold scheme (Otsu, 1975). In this binary image, '0' indicates the black pixel.

Thinning is not applied, as the stroke width of the characters from text in the wild varies from thin to thick strokes.

4.2 The feature types

The features used for the object recognition can be categorized into two types - global and local. Global features describe the global patterns in the entire image. Such features are useful because of their compact representation of images, where each feature corresponds to a point in a high-dimensional feature space. As a result, the data representation for the classifier is in the form of n-dimensional vectors in $\boldsymbol{R}^n$.

But, global features are sensitive to clutter and occlusion. They work better when an image contains only a single object, or when the segmentation of the object from the background is performed accurately. An alternative is to use local features, which are the descriptors of local image neighbourhoods computed at multiple points of interest. They may be used to recognize the object despite significant clutter and occlusion. They do not require segmentation of an object from the background, unlike many global features, or representations of the object's boundary (shape features). Typically, the points of interest are detected at multiple scales and different views of an object. The points try to capture the essence of the object's appearance. The local feature descriptor describes the image patch around a point of interest.

In order to make use of the existing tools of object category recognition to the character recognition, the following 6 types of feature representations are chosen for evaluation – Pixel density [3], Directional features [2], GIST [11], HOG [4], Dense SIFT [9] and Shape Context [1]. These features have been reported to work well on the object recognition [5, 7, 18] in combination with some popular classification approaches. These features and classifiers are not specific to the Devanagari but are found to be suitable for the English character recognition in natural scenes. It is important to study these feature representations to understand when and where they work. The following sections attempt to group the feature representations according to their type.

4.2.1 Global features

The global features represent the whole image as a single vector. This allows us to use classifications tools like SVM etc. A few global feature representations are now described.

Pixel density: The Pixel density [3] is derived from a character using zoning technique. First the character image is converted into the binary form by using Otsu's threshold method [12]. It is then normalized to the size of 32×32. This image is divided into 8×8 blocks, each of size 4 × 4. The ratio of the average pixel count of black pixels, representing the character, to the total number of pixels in the block is taken as a feature for each block. Thus there will be $\frac{32}{4}\times\frac{32}{4}=64$ features from each image.

Directional features: These are proposed in [2] for recognizing the English characters. They are geometric features derived from the character contour. They are of the line type that forms the character skeleton. The preprocessed image is divided into 9 equal sized windows called tones. Feature extraction is carried out from individual zones rather than the whole image. Every zone has the following 9 features: 1) Number of horizontal lines, 2) Number of vertical lines, 3) Number of Right diagonal lines, 4) Number of Left diagonal lines, 5) Normalized length of all horizontal lines, 6) Normalized length of all vertical lines, 7) Normalized length of all right diagonal lines, 8) Normalized length of all left diagonal lines, 9) Normalized area of the skeleton. Thus there are 81(9×9) features for each alphabet sample.

GIST: Oliva and Torralba [11] proposed GIST descriptor to represent the spatial envelope of the scene. The GIST descriptor of an image is the windowed 2D Gabor filter responses of an input image. The responses of Gabor filters encode the texture gradients that describe the local properties of the image. Averaging out these responses over larger spatial regions gives us a set of global properties. In the current work, the image is divided into 4×4 grids. Gabor filter responses at 2 scales and 2 orientations are collected. Thus the feature vector of each character image is of the order (4×4) × (2+2) = 64.

Histogram of Oriented Gradients **(HOG)**: HOG is devised by Dalal and Triggs [4] to overcome the problems associated with the features generated from raw pixels. This feature set is independent of the image size and captures the information about the intensity gradients. The window size and the number of bins in the histogram can be varied to analyze the performance of classification with respect to the HOG feature size. Thus, this provides a flexible set of representative features and helps to deal with both high bias and high variance issues. The image is divided into small spatial regions (or cells). A local 1-D histogram of the gradient directions or edge orientations over the pixels of each cell are accumulated. These histograms form the HOG features. A measure of local histogram energy over somewhat larger spatial regions (blocks) is used to normalize all the cells in the block. This helps achieve better invariance to illumination, shadowing, etc. A 4×4 size cell was considered to yield a 10-dimensional feature from each histogram. The overall feature vector is of size 4×4×10 = 160.

4.2.2 Local features

One of the key issues in dealing with the local features is that they vary in number for each image, thus making the matching more complicated. This problem is addressed by using a codebook to derive a single vector representation for an image, which allows the use of the standard classification techniques. However, deriving a global feature from the local patch descriptors using a codebook can lead to loss in the discrimination. Hence there is a trade-off in deriving a global representation from the local descriptors.

Scale Invariant Feature Transform (SIFT): SIFT features by Lowe [9] are evolved by locating the points of interest and then deriving the histograms of the gradient orientations computed around these points. Several approaches for the keypoint detection include the local maxima of the difference-of-Gaussians, Harris Hessian-Laplace detector, which gives affine transform parameters. The SIFT feature descriptor is a set of orientation histograms on (4 × 4) pixel neighbourhood. The histograms are partitioned into 8 bins each, and each descriptor is extracted from a 4 × 4 array of 16 histograms around the key-point. This leads to the feature vector of size 128.

As discussed before, the local maxima keypoints are mostly available for the textured regions. In this work, a variation of SIFT, namely Dense SIFT features [9] are used. These are derived by densely sampling keypoints from the character and extracting SIFT descriptors around them.

Shape Context [1]**:** This is a way of describing shapes by the measure of shape similarity. It is similar to the SIFT descriptor, but is collected around edges. Shape context is a 3D histogram of edge point locations and orientations. Edges are extracted by an edge detector. For each point on the edges, the coarse histogram of the relative coordinates of the remaining $n - 1$ points is taken as the descriptor. The descriptor is a log-polar histogram, which gives a $\theta \times r$ vector, where θ is the angular resolution and r is the radial resolution. We have used $\theta = 16$ and $r = 6$ giving rise to 96 length feature descriptor.

Local features from the images form a set of descriptors. To derive a global representation, a Bag of Features/Words (BOW) approach is employed by Sivic *et al.* [16]. This representation performs well on the objection recognition front.

5. RESULTS

We now describe the experiments carried out using 5 different types of classifiers (a) Nearest Neighbour (NN) classifier which uses χ^2 statistics as a similarity measure; (b) Support Vector Machines (SVM); (c) Naïve Bayes classifier; (4) C4.5 decision tree and (5) Boosted C4.5.

In this, the recognition performance of the popular feature representations discussed in Section 4.2 is evaluated using the above classifiers on different datasets described in Section 3.

5.1. Performance on DSMP-4K and DSHnd-30K

The features that have been reported to work well on the Roman script are investigated for their suitability in recognizing Devanagari script. Both DSMP-4K and DSHnd-30K datasets are considered for this test. 5-fold cross-validation is done to evaluate the usefulness of the feature representations and classifiers in terms of generalization ability. The results obtained are summarized in Table 3 (for DSMP-4K) and Table 4 (for DSHnd-30K).

In Table 3, most feature representations give good results for this data set. Note that the directional features which work well for the Roman script recognition does not perform well for the Devanagari script. This is also evident from Table 4 for the handwritten character recognition implying that different scripts require different feature representations. It is also interesting to note that the existing feature representations for the character recognition work very well for the machine printed character recognition, particularly GIST features which are mainly used for the scene matching. This is because the machine printed characters are more predictable in terms of their form and type and a model trained on the machine printed characters would perform well in recognizing the machine printed test characters. However, this is not the case with the handwritten character recognition.

We note that the local features like Shape Context and SIFT fare poorly when it comes to the handwritten character recognition on the DHnd-30K data set. This is due to the local variability of the handwritten text which is unpredictable. There are also some significant local dissimilarities between the characters of the same category due to the variations in the handwriting. In contrast, global feature representations fare better. GIST gives the best recognition rates over other features. The intuition is GIST being a global feature representation is robust to small local variations.

Table 3 5-fold Cross-validation rates obtained on DSMP-4K

Classifier Features	NN	SVM	Naïve Bayes	C4.5	Boosted C4.5
Pixel density	96.67	99.20	45.02	77.37	94.53
Directional feature	82.95	85.67	40.83	56.58	76.58
GIST	95.95	**99.78**	70.47	77.01	94.58
HOG	98.76	99.34	65.09	72.32	91.37

Table 4 5-fold Cross-validation rates obtained on DSHnd-30K

Classifier Features	NN	SVM	Naïve Bayes	C4.5	Boosted C4.5
Pixel density	61.39	76.56	48.12	44.05	61.67
Directional feature	33.14	59.97	42.48	24.55	37.75
GIST	61.4	**89.02**	61.51	52.02	73.30
HOG	71.99	86.03	61.05	49.62	66.91
Shape Context	37.43	60	40.21	29.51	39.87
Dense SIFT	40.5	61	53.65	43.32	50.19

5.2 Performance on the Transformed Dataset DSMP-24K

In the real life situation, most text can be expected to be distorted versions of the printed text. The recognition performance of features is studied over an affinely transformed data set of Devanagari characters. Although this dataset is synthetically generated it is reasonable to consider the cases where the test characters have undergone the unknown geometric transformation. Firstly, we find, whether a classifier trained on the machine printed fonts can identify the same fonts under affine transformations. The DSMP-4K dataset is used as the training data and the *transformed* data set DSMP-24K (as described in Section 3.1c) is taken as the testing set. The results obtained are summarized in Table 5. Secondly, 5-fold cross validations results are obtained on the DSMP-28K datasets. This is to find the generalization capability, given the transformed data for learning the character category model. The results are summarized in Table 6.

The highest recognition rate obtained on the DSMP-28K data set is 99.53% using GIST features and SVM classifier. But for the same combination of feature and classifier, the recognition rate is 70.59% on the *transformed* set. Similar drop in recognition rates can be observed for the remaining combinations. There is a deterioration of the results with the directional features in Table 4. The pixel density feature is found to decline from a highest value of 99.08% in Table 3 to a value of 55.04% in Table 5. This reveals the issues arising from recognizing the text characters taken from the real life scene images. Interestingly, the cross validation results are very good for the transformed data set used in Table 6 but lower than those on the DSMP-8K dataset which does not contain the transformed data. Comparison of these results with those in Table 5 clearly indicates that lack of representative training data in modelling a character category hampers the recognition rate of the unknown test data. Hence we need techniques having better generalization but requiring less training data.

Table 5 Classification rates obtained on DSMP-4K (training) and DSMP-24K(testing)

Classifier Features	NN	SVM	Naïve Bayes	C4.5	Boosted C4.5
Pixel density	55.04	49.44	30.67	23.38	33.74
Directional feature	26.17	39.04	25.64	18.46	28.31
GIST	48.8	**70.59**	52.79	30.61	46.32
HOG	79.09	62.80	30.44	22.48	39.56
Shape Context	34.56	46.19	37.70	24.29	31.35
Dense SIFT	35.13	47.92	36.88	22.26	29.42

Table 6 5-fold Cross-validation rates obtained on DSMP-28K

Classifier Features	NN	SVM	Naïve Bayes	C4.5	Boosted C4.5
Pixel density	96.12	99.08	74.33	77.68	94.11
Directional feature	79.57	88.39	58.44	59.05	80.80
GIST	95.31	**99.53**	90.89	75.97	94.48
HOG	98.68	99.04	89.17	76.25	95.23
Shape Context	84.62	89.89	77.34	59.51	73.89
Dense SIFT	81.45	90.83	82.61	65.39	77.68

5.3 Performance on text from the wild

The performance of the trained model (on the existing databases DSMP-28K and DSHnd-30K) is now studied on the characters segmented from the images of the real wild scenes (DSIW-3K). Results are summarized separately in Tables 7 and 8. The highest recognition rates observed is 46.38%

Smaller recognition rates are mainly due to the significant amount of noise and ambiguity associated with the characters DSIW-3K. Apart from the environmental noise, artistic

renditions (3D characters, shadows etc) and geometric transformations, the data also contains conjuncts, parts of *matras*, compound characters etc. An interesting observation is that the training on the machine printed or the handwritten samples doesn't seem to make a significant difference in the performance. But as the machine printed samples can be generated more easily than the handwritten samples, it is worth pursuing this approach.

In comparison to all other feature representations considered, GIST features give the most consistent recognition rates on all the classifiers considered revealing the fact that the global features are more reliable in recognizing the text from the images of the wild where the characters undergo transformations but their global appearance remains somewhat similar to the original.

5.4 Performance on DSIW-3K

The performance of the feature-classifier combinations are now evaluated on the DSIW-3K data set. Note that some of the character categories in this data set have very few samples (one in some categories) and the character samples are collected from the wild. The lack of training data affects the recognition rate.

Table 7 Classification rates obtained on DSIW-3K using DSMP-28K

Classifier Features	NN	SVM	Naïve Bayes	C4.5	Boosted C4.5
Pixel density	30.99	12.36	20.22	10.59	13.36
Directional feature	17.25	17.17	9.65	3.92	4.32
GIST	39.18	**46.38**	30.86	18.07	29.04
HOG	38.82	10.05	25.49	19.91	28.53
Shape Context	22.23	25.23	18.95	11.05	14.06
Dense SIFT	23.52	32.14	25.99	18.69	23.55

Table 8 Classification rates obtained on DSIW-3K using DSHnd-30K

Classifier Features	NN	SVM	Naïve Bayes	C4.5	Boosted C4.5
Pixel density	10.93	10.17	14.58	5.59	9.87
Directional feature	8.91	14.67	13.83	3.11	10.23
GIST	26.47	**30.52**	16.23	29.04	30.21
HOG	**33.48**	24.59	28.57	21.6	28.23
Shape Context	10.95	9.87	10.67	6.64	8.97
Dense SIFT	11.46	10.23	13.63	8.60	10.18

Table 9 gives the recognition rates obtained from the 5-fold cross validation tests on the DSIW-3K data set, using a Nearest Neighbour classifier. GIST features perform the best followed by the HOG features. Interestingly global features like pixel density perform better than the local features like SIFT. This is justified by the fact that given the amount of variability in the form and types of characters of this data set, global features give an overall representation that is better than that of the local features. Moreover, the local features are affected by the local noise and variability's of renditions.

Finally in Table 10, a state-of-the-art works in other languages is presented. As there is no Devanagari work so far in the literature, this work and the database can serve as baseline results for future researchers.

Table 9 5-fold Cross validation results on DSIW-3K

Classifier Features	NN
GIST	**56.03**
HOG	55.58
Pixel Density	44.83
Directional feature	21.32
Dense SIFT +BOW	30.19
Shape Context +BOW	23.04

Table 10 State-of-the-art performance on the character recognition in the wild

Ref.	Database	Language	Performance
[5]	Chars74K	English	55.26%
	Chars74K	Kannada	2.77%
[19]	Indigenous	Chinese	63.9%
[7]	Chars74K	English	57.5%
	ICDAR 2003	English	51.5%
[10]	Chars74K	English	71.6%
	ICDAR2003	English	67.0%
Proposed	DSIW-3K	Devanagari	56.03%

6. CONCLUSIONS

A database of Devanagari characters captured from the wild is created. The performance evaluation of the existing state-of-the-art features and classifiers on this database is presented. This database and results will serve as baseline results for the future researchers in this direction.

The existing OCR methods, although good for the scanned images of printed text, perform poorly on characters extracted from the images of the wild. This is because the images of the wild contain the unforeseen fonts, 3D effects and have distortion and noise in the characters. This motivates to develop some improved methods for recognizing the characters in the images of the wild, which have a wide range of applications with the advent of ubiquitous mobile technology.

Lack of training data for the character categories of the wild images is a serious bottleneck. Future works can attempt to develop a method that uses the synthesized Devanagari characters yet produce recognition rates closer or even better that that using the characters from the wild as the training data set.

7. ACKNOWLEDGMENTS

We are very grateful to Shorya Gupta for his contribution in the collection of the data.

8. REFERENCES

[1] Belongie, S., Malik, J., and Puzicha, J. 2001. Shape matching and object recognition using shape contexts. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol.24, pp.509-522

[2] Blumenstein M., Verma B., and Basli H. 2003. A novel feature extraction technique for the recognition of segmented handwritten characters. ICDAR, Vol. 1, pp.137-141.

[3] M. Bokser. 1992. Omnidocument technologies. Proc. IEEE (Jul. 1992), Vol. 80, No. 7, pp. 1066–1078.

[4] Dalal N. and Triggs B. 2005. Histograms of oriented gradients for human detection. IEEE Conf on Computer Vision and Pattern Recognition (CVPR), San Diego CA, June 20-25, 2005, pp.886-893.

[5] de Campos T., Babu B., and Varma M. 2009. Character recognition in natural images. International Conference on Computer Vision Theory and Applications, Lisbon, Portugal, (Feb 2009), pp.273-280.

[6] Kai Wang, Boris Babenko and Serge Belongie. 2011. End-to-End Scene Text Recognition. IEEE International Conference on Computer Vision (ICCV), Barcelona, Spain, 2011.

[7] Kai Wang and Serge Belongie. 2011. Word Spotting in the Wild. ECCV 2010, Tokyo.

[8] Liang J., Doermann D., and Huiping Li. 2005. Camera-based analysis of text and documents: a survey. International Journal on Document Analysis and Recognition 2005, Vol.7, No.2, pp.84-104.

[9] Lowe, D. G. 2004. Distinctive image features from scale-invariant keypoints. IJCV, 2004, Vol. 60, No. 2, pp.91-110.

[10] L Neumann and J Matas. 2011. A Method for Text Localization and Recognition in Real-World Images. LNCS, pp. 770–783.

[11] Oliva. A. and Torralba. A. 2001. Modeling the shape of the scene: a holistic representation of the spatial envelope. International Journal of Computer Vision, Vol.42, No.3, pp.145–175.

[12] Otsu N. 1979. A threshold selection method from gray level histograms. IEEE Transactions in Systems, Man and Cybernetics (March 1979), Vol. 9, pp.62-66.

[13] Pal U., Sharma N., Wakabayashi T., and Kimura F. 2007. Off-Line Handwritten Character Recognition of Devanagari Script. International Conference on Document Analysis and Recognition (ICDAR), 23-26 Sept. 2007, Vol. 1, pp.496 – 500.

[14] Pal U., Wakabayashi T., and Kimura F. 2009. Comparative Study of Devanagari Handwritten Character Recognition using Different Feature and Classifiers. International Conference on Document Analysis and Recognition (ICDAR), pp.1111-1115.

[15] Ramana Murthy O. V., and Hanmandlu M. 2011. A Study on the Effect of Outliers in Devanagari Character Recognition. International Journal of Computer Applications (October 2011), 32(10):10-17.

[16] Sivic J., Russell B. C., Efros A. A., Zisserman A., Freeman W. T. 2005. Discovering objects and their location in images. IEEE International Conference on Computer Vision (ICCV), pp.370-377.

[17] Vikas J Dongre and Vijay H Mankar. 2010. A Review of Research on Devanagari Character Recognition. International Journal of Computer Applications (December 2010), 12(2):8–15.

[18] Weinman, J. J., and Learned Miller E., 2006. Improving recognition of novel input with similarity. In Proc. IEEE Conf on Computer Vision and Pattern Recognition (CVPR), New York NY, Vol. 1, pp-308-315.

[19] Zheng Q., Chen K., Zhou Y., Gu C., and H Guan. 2010. Text Localization and Recognition in Complex Scenes Using Local Features. LNCS (Proc. ACCV 2010), Vol. 6494, pp. 121–132.