A Holistic Method to Recognize Characters in Natural Scenes

Muhammad Ali and Hassan Foroosh
Department of Electrical Engineering & Computer Science, University of Central Florida, Orlando, Florida, U.S.A.

Keywords: Natural Scene Text Recognition, Active Contours, Holistic Character Recognition.

Abstract: Local features like Histogram of Gradients (HoG), Shape Contexts (SC) etc. are normally used by research community concerned with text recognition in natural scene images. The main issue that comes with this approach is ad hoc rasterization of feature vector which can disturb global structural and spatial correlations while constructing feature vector. Moreover, such approaches, in general, don't take into account rotational invariance property that often leads to failed recognition in cases where characters occur in rotated positions in scene images. To address local feature dependency and rotation problems, we propose a novel holistic feature based on active contour model, aka snakes. Our feature vector is based on two variables, direction and distance, cumulatively traversed by each point as the initial circular contour evolves under the force field induced by the image. The initial contour design in conjunction with cross-correlation based similarity metric enables us to account for rotational variance in the character image. We use various datasets, including synthetic and natural scene character datasets, like Chars74K-Font, Chars74K-Image, and ICDAR2003 to compare results of our approach with several baseline methods and show better performance than methods based on local features (e.g. HoG). Our leave-random-one-out-cross validation yields even better recognition performance, justifying our approach of using holistic character recognition.

1 INTRODUCTION

Recognition of text in natural scene context is a challenging problem in computer vision, machine learning and image processing. The importance to solve this problem is equally compelling due to abundant availability of digital cameras, esp. those in mobile devices e.g. smartphones and wearable glasses (e.g. Google Glass). There is a need of applications like assisted navigation for visually impaired people (e.g. OrCam device mounted on glasses, www.orcam.com) on one hand and mining huge online image data repositories for textual content (to automatically generate useful information for marketing, archival, and other purposes etc.) on the other hand. Other utilities of natural scene text recognition include and automatic reading of informational signs for automobile drivers or driverless cars.

Although the problem of document text recognition is a solved one, yet its counterpart in natural scene scenario is far from being solved. This is primarily due to the difficult and potentially uncontrolled imaging conditions that occur while imaging text in the wild. Hence the problem has been broken down into the following four sub problems:

1. Cropped Character Recognition
2. Cropped Word Recognition
3. Scene Text Detection
4. Full-image Scene Text Recognition

There is another related problem proposed by (Wang et al., 2011) to recognize words in an image given a short vocabulary pertaining to the image.

To compare results, several public datasets are in use, e.g. ICDAR2003 (Lucas et al., 2003), WLM dataset (Weinman et al., 2009), Chars74K datasets (de Campos et al., 2009), SVT (Wang et al., 2011), NEOCR (Nagy et al., 2011), etc.

In this paper we address the sub problem 1 above and use the two most popular datasets like ICDAR2003 robust character dataset and the Chars74K datasets.

Figure 1 shows sample images from Chars74K and ICDAR datasets. The challenge is obvious from the extent of shape variation and sample noise due to distortions, illumination differences, object occlusions, etc. Hence, we are confronted with all sorts of structured and/or random noise.

Ali, M. and Foroosh, H.
A Holistic Method to Recognize Characters in Natural Scenes.
In *Proceedings of the 11th Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISIGRAPP 2016) - Volume 4: VISAPP*, pages 449-457
ISBN: 978-989-758-175-5

Figure 1: Sample characters from Chars74K and ICDAR datasets. Top row shows samples from synthetic Chars74K-Font dataset.

The paper is organized as follows: Section 2 describes the related research work done on this problem. Section 3 gives a brief look at active contour model used in this paper. Section 4 describes steps in our method to solve the problem. Section 5 presents experimental setup, results, and discussion. Section 6 gives a recap of this paper along with future research directions.

2 RELATED WORK

Since the introduction of ICDAR 2003 Robust Reading Competition and the associated challenge datasets (Lucas et al., 2003), the area of scene text recognition has seen an increase in research efforts to solve the problem. Various solutions have been proposed for the sub problem of robust character recognition.

Some researchers used off-the-shelf OCRs to recognize characters. (Chen and Yuille, 2004) used an adaptive version of Niblack's binarization algorithm (Niblack, 1985) on the detected textual regions and then employed commercial OCRs for final recognition. The reported results with ABBYY (www.abbyy.com) were good for their dataset (collected from cameras mounted on blind people.) Later performance of ABBYY reported by (Wang and Belongie, 2010) and (de Campos et al., 2009) showed its poor performance on more challenging ICDAR and Chars74K datasets.

Overall, the literature in natural scene character recognition is dominated by local feature-based methods: These methods mainly focus on extracting a feature vector, e.g. a Histogram of oriented Gradients (HoG) (Dalal and Triggs, 2005) or some variant of it, from a character image and then using some classifier, e.g. Nearest Neighbor, SVM etc., to recognize the character. (de Campos et al., 2009) used various feature descriptors including Shape Contexts (SC), Scale Invariant Feature Transform (SIFT), Geometric Blur (GB), etc. in combination with bag-of-visual-words model. The results, however, left a lot of room for improvement. (Weinman et al., 2009) used a probabilistic framework wherein they utilized Gabor filters in their similarity model to recognize characters in their dataset. (Wang and Belongie, 2010) showed better performance than (de Campos et al., 2009) by incorporating HoG features. (Neumann and Matas, 2011) used maximally stable extremal regions (MSER) to create MSER mask and then got features along its boundary which they used in SVM for classification. (Donoser et al., 2008) used MSERs in conjunction with simple template matching to get initial character recognition results which are subsequently improved by exploiting web search engines to get final recognition results. In addition, unsupervised feature learning system has been proposed by (Coates et al., 2011) that utilizes a variant of K-means clustering to first build a dictionary then map all character images to a new representation in the dictionary. A recent holistic approach based on tensor decomposition has been proposed in (Ali and Foroosh, 2015) that takes into account some problems with local feature based methods but the authors perform ad hoc pre-processing to account for rotation. Hence their method relies on making a character image upright to take care of rotated characters.

In this paper, we propose a holistic approach to solve the problem and present a novel feature vector based on active contour model. The use of active contour models in shape recognition is common but we are not aware of its application on scene text recognition. The closest application we found was in (Yi and Tian, 2014) where the authors establish boundary points using discrete contour evolution during the process of finding character polygons as a first step in getting stroke configurations. Other than this, their approach quite different from ours.

The results we get show that the proposed method effectively captures character shape variations occurring in natural scene images in a holistic manner thus avoiding the problems associated with ad hoc rasterisation of local image features. Our contributions are:

1. Novel feature vector
2. Rotation invariance

Our results show that we perform better than several popular baseline methods.

3 SNAKE: AN ACTIVE CONTOUR MODEL

As described above, we are interested in holistic recognition of characters extracted from natural scene images. An active contour is a dynamic object (curve) which evolves to wrap around an object boundary. This idea of capturing object shape motivates us to use it to extract holistic feature for our character recognition problem.

Active contour models have been used for image segmentation and shape description (Kass et al., 1987; Xu and Prince, 1997; 1998) since long time. In the following section, we first briefly recap the basics and then move on to give its novel application to deriving our holistic feature for characters synthetic or extracted from natural scene images.

3.1 Basics

Consider the situation of a 2D image. An active contour model can be described as a closed loop of points or pixels. It is also called a snake for its movement in image plane. Mathematically, a snake is a set of points in the image plane. It can be characterised by the following parametric curve:

$$u(s) = \big(x(s), y(s)\big) \tag{1}$$

where, x and y are the coordinates of pixels and s is the parameter. As described in (Ivins and Porrill, 2000), we can associate an energy functional E with this curve (note that the curve is a loop in this case).

$$E(u) = \oint P(u) + \alpha(s)|u'|^2 + \beta(s)|u''|^2 \, ds \tag{2}$$

where, $P(u)$ is the external image energy (derived mainly from image gradients) and α and β are the internal curve parameters for tension and stiffness respectively.

Assuming $\alpha(s) = \boldsymbol{\alpha}$, and $\beta(s) = \boldsymbol{\beta}$ as constants, the minimization of equation (2) can be done by satisfying two independent Euler equations in u:

$$\boldsymbol{\beta} u'''' - \boldsymbol{\alpha} u'' = -\frac{d(P)}{du} \tag{3}$$

Following (Ivins and Porrill, 2000), the derivatives in (3) can be approximated by finite differences as follows:

$$\boldsymbol{\beta}(x_{s-2} - 4x_{s-1} + 6x_s - 4x_{s+1} + x_{s+2}) - \boldsymbol{\alpha}(x_{s+2} + x_{s-2} - 2x_s) = f_x(x, y) \tag{4}$$

$$\boldsymbol{\beta}(y_{s-2} - 4y_{s-1} + 6y_s - 4y_{s+1} + y_{s+2}) - \boldsymbol{\alpha}(y_{s+2} + y_{s-2} - 2y_s) = f_y(x, y) \tag{5}$$

where f_x and f_y are the components of image force (computed from gradients in 'x' and 'y' direction). To solve (4) and (5), we use the semi-implicit method discussed in (Ivins and Porrill, 2000) where two sets of finite difference equations are formed to describe the x and y coordinates of the entire snake; these equations can be written in terms of a *cyclic symmetric pentadiagonal banded matrix* $\mathbf{M}$ incorporating the constants $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ as follows:

$$\mathbf{M}.\mathbf{x} = \mathbf{f}_x(\mathbf{x}, \mathbf{y}) \;\; ; \; \mathbf{M}.\mathbf{y} = \mathbf{f}_y(\mathbf{x}, \mathbf{y}) \tag{6}$$

where, $\mathbf{x}$ and $\mathbf{y}$ are vectors containing the x and y coordinates of all the snake elements; $\mathbf{f}_x$ and $\mathbf{f}_y$ are the corresponding vectors of image forces acting on contour points. We can solve these equations iteratively by using a discrete and small time step 'τ' (we set $\tau = 1$ for our experiments). The stiffness and tension constraints are applied at time t+1 after adjusting the snake according to the image forces at time t:

$$\mathbf{x}_{t+1} = (\mathbf{M} + \tau\mathbf{I})^{-1}.(\mathbf{x}^t + \tau\mathbf{f}_x(\mathbf{x}^t, \mathbf{y}^t) \tag{7}$$

$$\mathbf{y}_{t+1} = (\mathbf{M} + \tau\mathbf{I})^{-1}.(\mathbf{y}^t + \tau\mathbf{f}_y(\mathbf{x}^t, \mathbf{y}^t) \tag{8}$$

The matrix inversion in the above equations (7 and 8) is taken only once because they are composed of constant terms. Hence, the above iterative minimization provides for a fast way to solve the equation (2).

We compute the external image energy, $P(u)$, by taking a weighted combination of three factors: lines, edges, and corners. Thereafter, we compute the effects of external forces on each point of the contour by interpolation. The new coordinates of contour points are computed from equations (7) and (8) above. To further expand the reach of image forces and let the snake enter concave regions, we also utilize gradient vector flow as mentioned in (Xu and Prince, 1998).

3.2 Feature Vector

As we evolve the snake around character images and compute new coordinates of each point at each time step of the above iterative minimization, we compute the following:

$$\Delta d_t = \sqrt{\Delta {x_t}^2 + \Delta {y_t}^2} \tag{9}$$

$$\Delta\theta_t = tan^{-1}\left(\frac{\Delta y_t}{\Delta x_t}\right) \tag{10}$$

where, $\Delta x_t = x_t - x_{t-1}$ and $\Delta y_t = y_t - y_{t-1}$. The quantities Δd_t *and* $\Delta\theta_t$ represent the distance and angle increments respectively for each point (pixel)

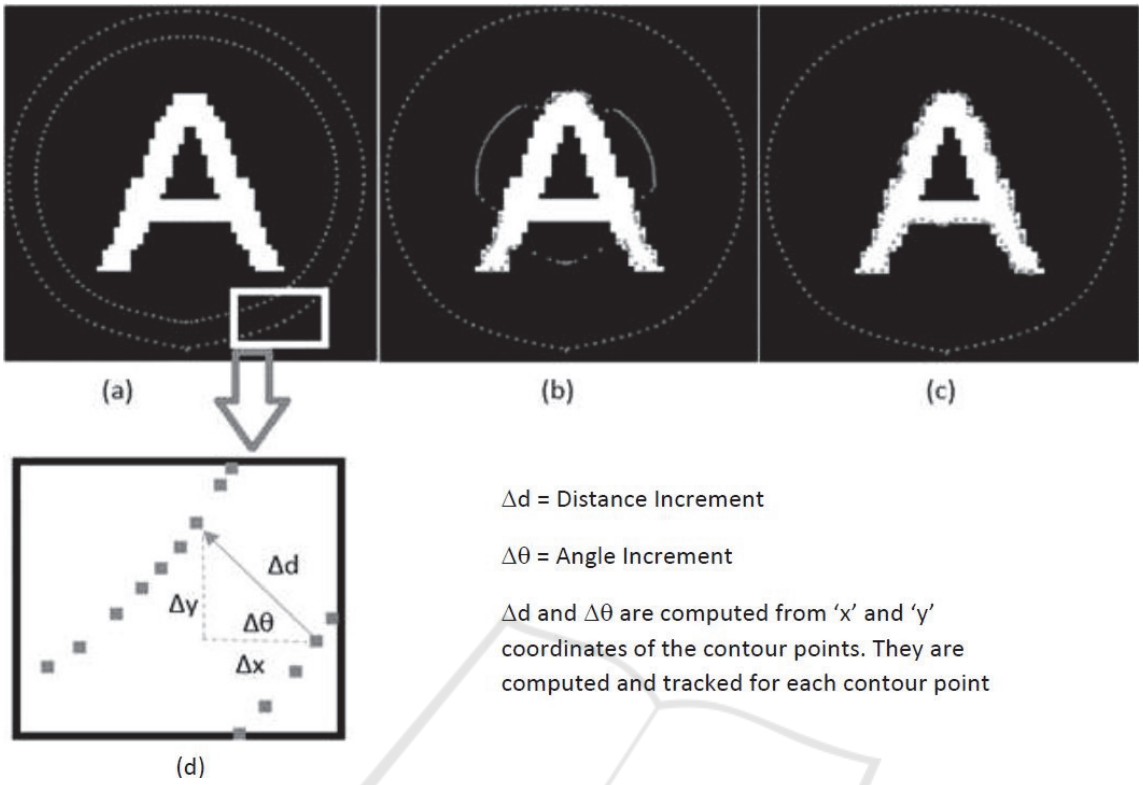

Figure 2: Illustration of our framework for feature vector extraction. Outer loop shows the location of pixels in the initial contour and inner loops show the evolution of the initial contour after some iterations. (a) Contour evolution after 50 iterations (b) After 125 iterations (note the wrapping of contour around the character (c) Final contour after 200 iterations (d) Enlarged view of the points and computation of distance and angle increments.

on the contour at time instant t. We accumulate the increments over the course of evolution for all points p and concatenate to form our descriptor as follows:

$$f(im) = \begin{pmatrix} (\sum_{t=1}^{n} \Delta d_t) p_1 \\ \cdot \\ \cdot \\ (\sum_{t=1}^{n} \Delta d_t) p_m \\ (\sum_{t=1}^{n} \Delta \theta_t) p_1 \\ \cdot \\ \cdot \\ (\sum_{t=1}^{n} \Delta \theta_t) p_m \end{pmatrix} \quad (11)$$

where, $f(im)$ is the final feature vector, and n is the number of iterations of the contour evolution. The size of the above feature vector is equal to twice the number of contour points m (i.e., $2m = 125 \times 2 = 250$ in our experiments).

The rotational invariance property of the above feature vector follows from its design (the initial contour is circular) and its use in conjunction with the cross-correlation similarity metric.

4 SCENE CHARACTER RECOGNITION

Our framework for scene character recognition starts with pre-processing images, followed by training where we generate feature vector using active contour evolution, as depicted in Figure 2, for all training samples, and finally classification by getting the feature vector for each test image and computing similarity using cross-correlation.

4.1 Pre-processing

The cropped character images from natural scene datasets contain a lot of non-character structures and imperfect cropping artefacts which makes it difficult to effectively capture typeface and shape variations. Since the focus of our work is to demonstrate effectiveness of holistic recognition framework based on active contour feature, we pre-process each image in training and testing sets to keep the images as noise free as possible. To this end we adopt binarization for

image segmentation to reduce noise and extract, possibly only, character structures. This somehow lets us isolate the classification problem from the binarization problem.

4.1.1 Image Segmentation & Normalization

Segmentation of textual information from natural scene images is a challenging problem due to noise and distortions introduced mainly by uncontrolled imaging conditions. Many researchers have attempted to tackle it, e.g. see (Chen and Yuille, 2004); (Mishra et al., 2011); (Kita and Wakahara, 2010); (Field and Learned-Miller, 2013).
Since the focus of our paper is character recognition, we adopt and extend the method in (Ali and Foroosh, 2015) in an effort to segment each image to get the correct textual foreground (in white). To this end, we obtain two binary images: one from the output of Otsu's method and the other by inverting it. At this point, we do a simple analysis of the skeleton by counting number of pixels of both images to get the correct segmented image. We then perform a connected component analysis based on the observation that cropped characters mostly fall in the middle of the image. We consider any small pixel group as noise if its size is less than a small fraction (<5%) of the size of the largest central connected component. The process is shown as a flow chart in the Figure 3.

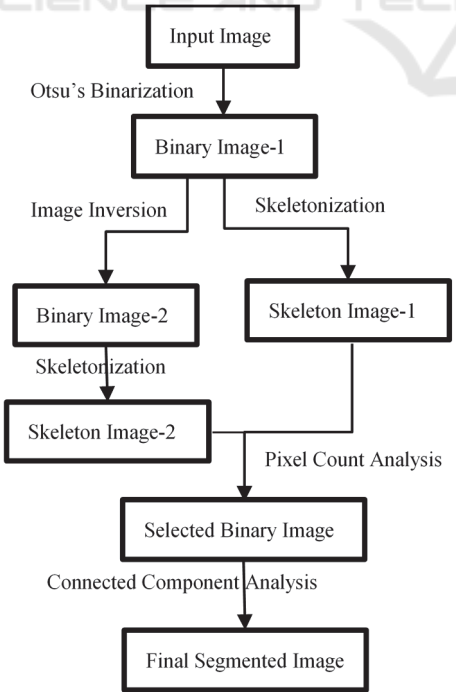

Figure 3: Segmentation of character images.

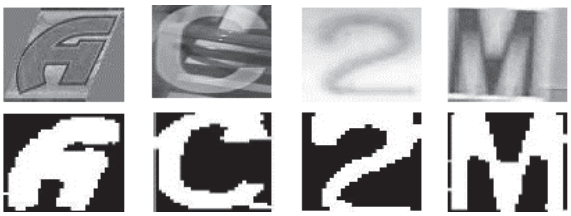

Figure 4: Sample binarized character images.

After segmentation, we normalize each image to have the size of 32x32 pixels. To make sure that curve evolution doesn't start too close to the character boundary, we pad the image with a 16x16 frame of zeros. Hence the final size of the image becomes 64x64.

4.2 Training

For training, we take individual images of each class and pre-process them. For each image, we envelope the character with a circular contour (radius of contour is fixed at 30 pixels) centred on the image and sampled uniformly with 125 points. The contour is then evolved towards the character and for each point on it, we accumulate direction (angle) and distance, until the character boundary is reached. The two quantities (direction and distance) are then concatenated and normalized to form a feature vector containing 250 elements.

Figure 2 illustrates the process of extracting the feature vector for a random image in the training process. The sensitivity of results for different parameter settings is discussed in (Section 5.3.1).

4.3 Classification

To classify a test image, we first pre-process it and then compute its feature vector by evolving a circular contour towards it. We then measure similarity of the test vector with each of the training vectors using the cross-correlation metric and recording the maximum. The final classification is given by taking the maximum over all classes.

5 EXPERIMENTS

We evaluated our approach using various character datasets. This includes synthetic dataset Chars 74K-Font as well as popular natural scene character datasets Chars74K-Image and ICDAR. We performed experiments using different settings to report our results on the datasets. We also compare our method with several baseline methods.

5.1 Datasets

The Chars74K-Font is a synthetic dataset of English alphabet generated in various typefaces for 62 character classes: 'A' to 'Z', 'a' to 'z', and digits '0' to '9'. The dataset consists of 62,992 images with 1,016 images per class.

The English subset of Chars74K dataset consists of 12,503 characters. Characters have been cropped from 1,922 images of advertisement signs and products from stores etc. The dataset is not split in training and testing sets, rather the authors (de Campos et al., 2009) give their proposed training and testing splits for comparison with their results. There is, however, a split between '*GoodImg*' and '*BadImg*' and as obvious from the names, the respective splits contain 'good' and less noisy (7705 images) as well as 'bad' more noisy images (4798 images) for a total of 12,503 images.

The ICDAR2003 robust character dataset contains 11,615 images of cropped scene characters and the dataset comes split into training and testing subsets. Characters have mostly been cropped from images of books titles, storefronts and signs and exhibit great variability in terms of resolution, illumination, colour, etc. The test set has 5340 images in total but those belonging to 62 classes (A-Z, a-z, and 0-9) are just 5,379.

5.2 Results

For all experiments, we fixed the parameters alpha and beta to 0.05, the number of iterations to 200, the radius of initial contour to 30 pixels. This parameter setting yield good results across all datasets. Further discussion on this setting is deferred until Section 5.3.1.

In the first experiment, we used Chars74K-Font synthetic font dataset. We randomly picked 15 samples each for training and testing to make fair comparison with the results of de Campos et al. The results are shown in Table 1. The second column of Table 1 presents interesting results when training on synthetic fonts and testing on Chars74K-Image test split proposed in de Campos et al. Here also, we show better performance than the reported method.

In our second experiment, we used the whole ICDAR2003 training set to get features for each class of characters. The accuracy on the test set was 62% (see Table 2).

(Wang et al., 2012) reported accuracy of 83.9% on a modified version of the ICDAR2003 test set, but they re-cropped all images for their experiments and their set contains 5198 images, which is less than those in ICDAR2003 test set. Hence, their results are not compared here.

In Figure 5, the lines parallel to the main diagonal of the confusion matrix reflect ambiguities due to character case, e.g. small case 'c' confused with 'C', etc. In Table 2, we also report our results on the training and test splits proposed by (de Campos et al., 2009) for Chars74K, viz., Chars74K-15, where the suffix '15' specifies the number of training and test samples to be used for the experiment.

Table 1: Character Recognition performance on Chars74K-Font dataset & Chars74K-15 Test Split.

Method	Chars74K-Font	Chars74K-15 Test Split
GB+NN (de Campos et al., 2009)	69.71%	47.16%
Proposed Method	**71%**	**56%**

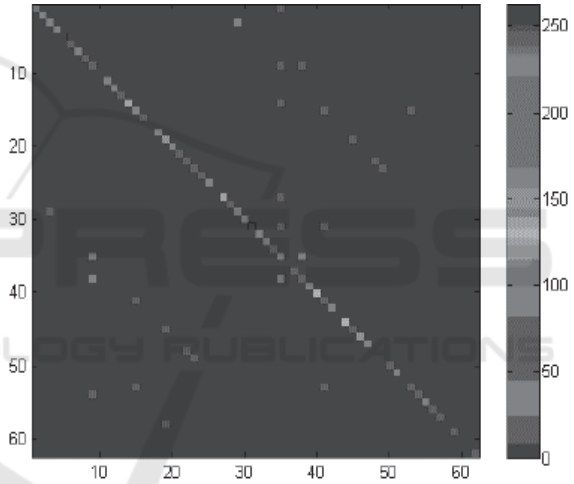

Figure 5: Confusion matrix for ICDAR2003 test set Numbers 1-62 show character classes A-Z, a-z,0-9. Lines parallel to the main diagonal show character confusions.

Table 2: Character Recognition performance on ICDAR2003 and Chars74K-15 datasets.

Method	ICDAR	Chars74K-15
GB+NN (de Campos et al., 2009)	41%	47.09%
HoG+NN (Wang and Belongie, 2010)	51.5%	58%
SYNTH+FERNS (Wang et al., 2011)	52%	47%
NATIVE+FERNS (Wang et al., 2011)	64%	54%
Stroke Config. (Yi and Tian, 2014)	62.8%	60%
Proposed Method	**62%**	**59%**

Our third experiment was to test the performance of Chars74K-15 test split on a modified training set (using all training samples but those in the test split) as per de Campos et al., The results are reported in Table 3.

Table 3: Recognition Performance on Chars74K-15 Test Split.

Method	Chars74K-15 Test Split
ABBYY FineReader (www.abbyy.com)	31%
GB+NN (de Campos et al., 2009)	54.3%
Proposed Method	**61.5%**

The difference in results of Table 2 (2nd column) and Table 3 clearly show that the number of training samples in the former (15 in this case) is not sufficient to capture the variation in test samples. Hence, we were prompted to do another experiment with leave-random-one-out cross-validation (CV) setting. We show results in Table 4 for both Chars74K and ICDAR2003.

For ICDAR2003 we combined training and testing sets to get one big set for CV. For Chars74K, as mentioned before, the data already comes without training and testing splits. The results in Table 4 show median accuracy over 100 trials.

Table 4: Recognition Performance using leave-random-one-out cross-validation (CV).

Method	ICDAR	Chars74K
Proposed Method + CV	65%	62%

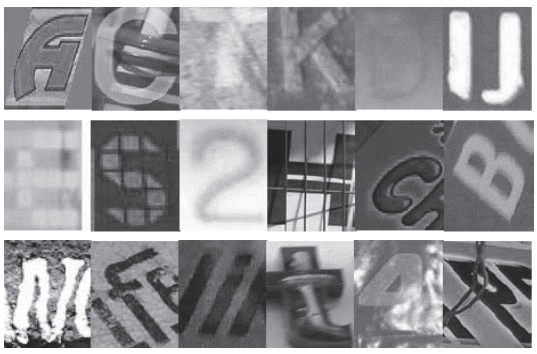

Figure 6: Some test samples from different datasets that our approach correctly recognized.

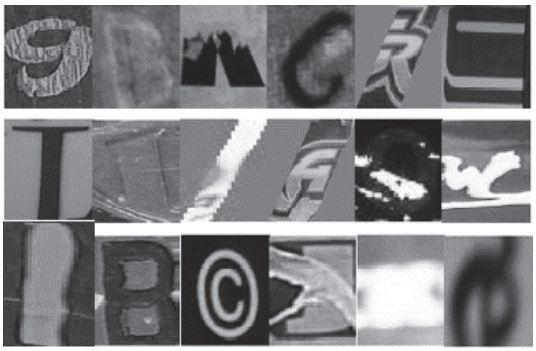

Figure 7: shows the cases where our method failed. Some images here are not even easy human observers to recognize correctly due to low contrast, shape ambiguities, noise etc.

5.3 Discussion

5.3.1 Parameter Sensitivity

In Figure 8, we show how the accuracy changes with respect to the snake's parameter values. The parameters α and β are internal smoothness coefficients and control how the contour behaves as it evolves over the given generations. .Here we assume α=β.

To estimate the values of the parameters, we perform experiment on the Chars74K-Font dataset and compute accuracy over different values of α and β spread over an arithmetic scale from 0.001 to 0.01. We found that the best accuracy occurs at α=β = 0.05. Hence, we use this value throughout our experiments.

The other factors influencing the results are noise. Although we segment the text in pre-processing step, yet we find that in many cases the reason of our system not performing well is the noise. We see that performance improvement can be achieved if a more elaborate algorithm is used for noise removal.

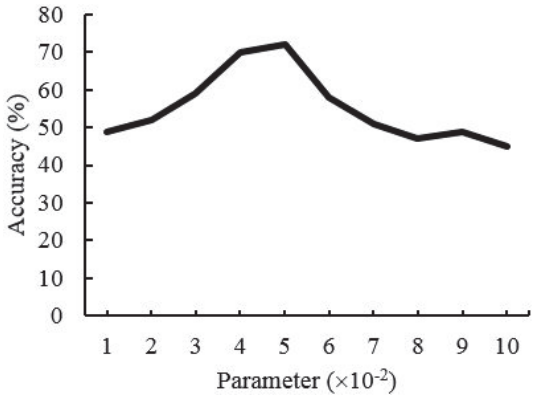

Figure 8: Snake internal parameters' sensitivity estimated on Chars74K-Font dataset. We assume parameters α=β in this case and the best value occurs at α = β = 0.05.

5.3.2 Runtime Analysis

The runtime of our approach can be estimated from the constituent processes namely, preprocessing and feature extraction. The preprocessing phase depends on Otsu binarization, and morphological skeletonizing operators, which are in general $O(n^2)$, where n is the size of the image. As regards the feature extraction phase, we deal with p contour points laid out in a circle. Each movement of the contour depends on computing external and internal forces acting on each point. Luckily, the iterative optimization method used in modelling snake evolution uses just one computation (involving matrix inversion) of internal force matrix whose size depends on p. External image forces involve computation of image gradients in horizontal and vertical directions and need $O(n^2)$ operations in one pass. Finally, we iterate over g evolution steps to get the snake to its terminal shape. Since, p and g are fixed prior to running the algorithm and p is usually very small compared with the size of image, the total cost turns out to be $O(n^2)$.

6 CONCLUSIONS

In this paper we put forth a novel feature to holistically solve natural scene character recognition problem that avoids dependency on specific features. Through our results we showed the potential of using our novel feature to better capture shape and font variations in scene character images. We got better results than several baseline methods and achieved improved recognition performance on the datasets using leave-random-one-out cross-validation, showing the importance of feature-independency and preservation of spatial correlations in recognition.

In future we hope to get state-of-the-art performance using better image segmentation methods and also optimizing other parameters of contour evolution. We also look forward to using contour evolution on grayscale images directly.

REFERENCES

Ali, M., and Foroosh, H., 2015. Natural Scene Character Recognition without Dependency on Specific Features. In *Proceedings of the International Conference on Computer Vision Theory and Applications (VISAPP), Berlin, Germany*, March 2015.

Chen, X., and Yuille, A., 2004. Detecting and reading text in natural scenes. In *Computer Vision and Pattern Recognition, 2004. CVPR 2004.* IEEE 2004. Vol. 2. pp. II–366.

Coates, A., Carpenter, B., Case, C., Satheesh, S., Suresh, B., Wang, T., Wu, D., and Ng, A., 2011. Text detection and character recognition in scene images with unsupervised feature learning. In *International Conference on Document Analysis and Recognition (ICDAR), 2011*. IEEE 2011, pp. 440–445.

Dalal, N., and Triggs, B., 2005. Histograms of oriented gradients for human detection. In *International Conference on Computer Vision and Pattern Recognition (CVPR) 2005.* IEEE 2005, pp.886-893.

de Campos, T. E., Babu, B. R., and Varma, M., 2009. Character recognition in natural images. In *Proceedings of the International Conference on Computer Vision Theory and Applications (VISAPP), Lisbon, Portugal, February 2009.* pp. 273–280.

Donoser, M., Bischof, H., and Wagner, S., 2008. Using web search engines to improve text recognition. In *19th International Conference on Pattern Recognition, ICPR 2008.* Vol. no. 14, pp. 8-11.

Hazan, T., Polak, S., and Shashua, A., 2005. Sparse Image Coding using a 3D Non-negative Tensor Factorization. In *International Conference on Computer Vision (ICCV), 2005.* IEEE 2005. Vol. 1, pp. 50–57.

Field, J., and Learned-Miller, E., 2013. Improving Open-Vocabulary Scene Text Recognition. In *International Conference on Document Analysis and Recognition (ICDAR) 2013.* IEEE 2013, pp. 604–608.

Ivins, J., and Porrill J., 2000. Everything you always wanted to know about snakes. *AIVRU Technical Memo 86, July 1993 (Revised June 1995; March 2000).*

Kass, M., Witkin, A., and Terzopoulos, D. 1987. Snakes: Active contour models. *International Journal of Computer Vision*. v. 1, n. 4, pp. 321-331.

Kita, K., and Wakahara, T., 2010. Binarization of color characters in scene images using k-means clustering and support vector machines. In *International Conference on Pattern Recognition (ICPR), 2010.* IEEE 2010, pp. 3183–3186.

Lucas, S. M., Panaretos, A., Sosa, L., Tang, A., Wong, S., and Young, R., 2003. ICDAR 2003 robust reading competitions. In *Proceedings of the Seventh International Conference on Document Analysis and Recognition 2003.* IEEE 2003. Vol. 2, pp. 682–687.

Mishra, A., Alahari, K., and Jawahar, C., 2011. An MRF model for binarization of natural scene text. In *International Conference on Document Analysis and Recognition (ICDAR), 2011*. IEEE 2011, pp. 11–16.

Nagy, R., Dicker, A., and Meyer-Wegener, K., 2011. NEOCR: A Configurable Dataset for Natural Image Text Recognition. In *CBDAR Workshop*, ICDAR 2011, pp. 53-58.

Neumann, L., and Matas, J., 2011. A method for text localization and recognition in real-world images. In *Computer Vision–ACCV 2010*, pp. 770–783.

Niblack, W., 1985. An introduction to digital image processing. Strandberg Publishing Company.

Otsu, N., 1979. A Threshold Selection Method from Gray-Level Histogram. In *Trans. System, Man and*

Cybernetics. IEEE 1979. Vol.9, pp.62-69.

Wang, T., Wu, D., Coates, A., and Ng, A., 2012. End-to-End Text Recognition with Convolutional Neural Networks. In *International Conference on Pattern Recognition (ICPR), 2012*. IEEE 2012, pp. 330.

Wang, K., Babenko, B., and Belongie, S., 2011. End-to-end scene text recognition. In *International Conference Computer Vision (ICCV), 2011*. IEEE 2011, pp. 1457–1464.

Wang, K., and Belongie, S., 2010. Word spotting in the wild. In *Computer Vision–ECCV 2010*, pp. 591–604.

Weinman, J., Learned-Miller, E., and Hanson, A., 2009. Scene text recognition using similarity and a lexicon with sparse belief propagation. In *Pattern Analysis and Machine Intelligence TPAMI*. IEEE Transactions 2009. Vol. 31, no. 10, pp. 1733–1746.

Xu, C., and Prince, J. L., 1998. Snakes, Shape, and Gradient Vector Flow. *IEEE Transactions on Image Processing*, 1998.

Xu, C., and Prince, J. L., 1997. Gradient Vector Flow: A New External Force for Snakes. In Proc. *IEEE Conf. on Comp. Vis. Patt. Recog. (CVPR)*, Los Alamitos: Comp. Soc. Press, pp. 66-71.

Yi, C., and Tian, Y., 2014. Scene text recognition in mobile applications by character descriptor and structure configuration, *IEEE Trans. IP*, pp 2972-2982, 2014.