reCAPTCHA assisted OCR for Devanagiri Texts

Kailash Atal ATALKAILASH@GMAIL.COM
Ashish Arora ASHISH.ARORA.IITG@GMAIL.COM
Indian Institute of Technology Guwahati, Assam, India, 781039.

Devendra Singh Sachan, Prof. P. K. Bora, Dr. Amit Sethi DEVENDRA.SACHAN@GMAIL.COM;PRABIN@IITG.ERNET.IN
Institute of Technology Guwahati, Assam, India, 781039. AMITSETHI@IITG.ERNET.IN

Abstract

CAPTCHA is a challenge response implemented using distorted characters on web to determine whether a user is a human or a computer. reCAPTCHA is constructive use of this human effort to digitize text from old documents which is difficult for OCR and also authenticate human user. The other paradigm is an integrated OCR-reCAPTCHA system where OCR digitizes documents with high accuracy and comes up with a confidence score so that characters with low score are digitized using human response. This work is first attempt to build reCAPTCHA assisted OCR system for Devanagiri script that learns from human response on the web and digitizes document with high accuracy.

1. Introduction

The problem of character recognition in scanned digital Hindi texts has received substantial attention from research community to assist digitization of a magazine article or a document using a scanner and OCR software.

From template matching, explicit handcrafted spatial features such as zoning, frequency domain features etc., the OCR systems moved to neural networks described by Lecun et al. We have implemented a new technique of deep learning that provides compact representation for diverse dataset of alphabets in Devanagiri script.

reCAPTCHA displays two strings: a computer generated string of random characters along with a word from scanned document so that a computer cannot overcome this challenge while a human giving correct response to computer generated string would very likely also enter the characters in the image segmented from the scanned document accurately. The word taken from a scanned document is digitized using majority response.

Appearing in Proceedings of the 1st Indian Workshop on Machine Learning, IIT Kanpur, India, 2013.

2. System Architecture

The proposed system contains two major blocks: OCR and reCAPTCHA system. The OCR system consists of document segmentation into individual characters and character recognition. A scanned document is segmented into lines, each line is segmented into words and words are segmented into individual characters based on projection profile described by Jawahar et al.

2.1 Character Recognition

Each character in the document is isolated into a 32x32 image. We train two layer convolution neural network (CNN) for feature learning integrated with L2 SVM for classification as described by Wang et al and shown in Figure 1. The classifier is trained for 48 complete Devanagari characters.

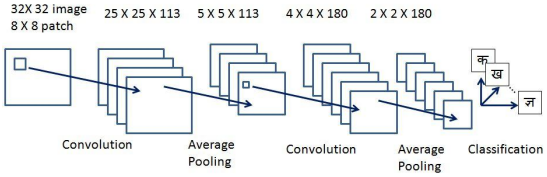

Figure 1. CNN L2-SVM architecture for Devanagiri OCR.

The image patches each of size 8×8 are extracted randomly from the training set of 32×32 images. Each 8×8 image patch $x^{(i)} \in R^{64}$ can be regarded as a 64 dimensional vector of pixel intensity values. The character images are normalized followed by zero phase component analysis (ZCA) whitening to obtain $x^{(i)}{}_{ZCA}$. A variant of K means clustering as described by Wang et al is used to learn basis D in the first layer of the CNN. The columns of matrix 'D' represent low level features such as strokes in the characters shown in Figure 2.

'D' is used to compute the activated response $z^{(i)} \in R^{d_1 x 1}$. By evaluating the activation over each 8×8 window in convolutional manner, we thus obtain $25\times25\times d_1$ dimensional representation of the input image.

$$z^{(i)} = \max\{0, |D^T x^{(i)}_{ZCA}| - \alpha\} \in R^{d_1} \quad \text{where } \alpha \text{ is constant} \qquad (1)$$

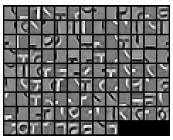

Figure 2. Basis vectors learnt from ZCA whitened images using a variant of K-means clustering.

Average pooling is performed over non-overlapping 5×5 hidden layer response ($z^{(i)}$) to get activation of a unit of pooling layer. We thus obtain a $5\times5\times d_1$ response map. To train the second layer, we similarly do convolutional feature extraction with a set of $2\times2\times d_1\times d_2$ weights from the first pooling layer to obtain $4\times4\times d_2$ response map followed by average pooling to obtain $2\times2\times d_2$ response where α=0.5, d_1=113 and d_2=180.

We use L2 SVM to predict class for a character image. Given a training set with input $z^{(i)} \in 2x2xd_2$ and output label $y^{(i)} \in \{1,-1\}$, the L2 SVM optimization problem as described in [3] is:

$$\min_{w,b} J = \| w \|_2^2 + \frac{C}{2}\sum_{m=1}^{M}[\max\{0, 1 - y^{(m)}(w^T z^{(m)} + b)\}]^2 \qquad (2)$$

We then learn the parameters 'w' and 'b' by batch gradient descent on J and fine-tune the weights of the second layer hidden units by error backpropagation on equation (2) in every iteration of gradient descent.

2.2 OCR-reCAPTCHA integrated System

The fusion of the reCAPTCHA and OCR is shown in Figure 3. We use the largest (d_{max}) and second largest (d_{max-1}) distance of the $2x2xd_2$ dimensional feature vector with all the hyperplanes to quantify the confidence of the OCR system to recognize the character correctly. A character is not recognized by the OCR system if:

$$d_{\max} \le p \text{ and } d_{\max} - d_{\max-1} \le q \qquad (3)$$

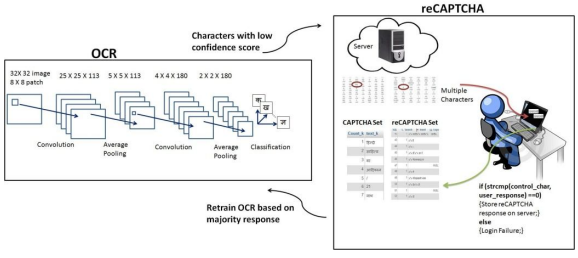

Figure 3. reCAPTCHA assisted learning based OCR system.(Accessible on : www.iitg.ernet.in/stud/k.atal/index.php)

p and q are obtained by maximising the ratio of increment in number of misclassified images to the increment in number of total images which satisfy equation (3).

Table 1. Database used for training and testing module for character recognition.

DATA SET	CHARACTERS USED FOR TRAINING	CHARACTERS USED FOR TESTING	ACCURACY
OCR ONLY	2400	24481	99.53
OCR INTEGRATED WITH RECAPTCHA (AFTER 1 DAY)	2562	24319	99.61
OCR INTEGRATED WITH RECAPTCHA (AFTER 2 DAYS)	2562	24319	99.75

3. Results

The accuracy of character recognition on uniform font printed characters using OCR has been calculated to be 99.53%. A total of 113 characters were misclassified from the pool of 24,481 characters in testing set. The error in recognition can be attributed to different characters which appear structurally very similar as shown in Figure 4.

Figure 4. Character misclassification using OCR system.

The accuracy of the reCAPTCHA system implemented for characters in Devanagari has been found to be 96.29%. More than 300 university students participated in this test. One primary reason for a marginally smaller accuracy of reCAPTCHA is due to the lack of ease for users on the web to enter Devanagari characters.

The value of p and q were obtained to be 0.02 and 0.56 respectively. We obtain a total of 162 character images out of 24,481 with low confidence. Out of 162, 70 characters were misclassified. These 162 images are digitized using reCAPTCHA. Using annotated labels based on majority user feedback, we retrain the OCR system with these additional 162 characters (Table 1).

References

Jawahar, C.V., Kumar, M. N. S. S. K. P., and Kiran, S. S. R. *A bilingual OCR for hindi-telugu documents*. Technical Report TR-CVIT-22, IIIT, Hyderabad, 2002.

Lecun, Y., Bottou, L., Bengio, Y., and Haffner, P. Gradient-Based Learning Applied to Document Recognition. Proceedings of the IEEE, Vol. 86, no. 11, 1998.

Wang, T., Wu, D.J., Coates, A., and Ng, A.Y. *End to End Text Recognition with Convolutional Neural Networks*. 21st International Conference on Pattern Recognition (ICPR), pp 3304-3308, 2012.