Automatic Detection of Handwritten Texts from Video Frames of Lectures

Purnendu Banerjee
Society for Natural Language Technology Research
Module 130, SDF Building
Kolkata-700091, India
purnendu@nltr.org

Ujjwal Bhattacharya and Bidyut B. Chaudhuri
Computer Vision and Pattern Recognition Unit
Indian Statistical Institute
203, B. T. Road, Kolkata-7000108, India
ujjwal@isical.ac.in, bbc@isical.ac.in

***Abstract*—Automatic recognition of handwritten texts in video lectures has important applications. In video lectures, the presenter usually writes on white / colored board. The video camera often captures the writing board along with certain other objects possibly including the presenter itself. Recognition of handwritten texts from such a video frame requires prior detection of the region of texts in the frame. In this article, we present our recent study of text localization in such video lecture frames. Here, we use Scale Invariant Feature Transform (SIFT) descriptors densely over the entire region of the frame. The descriptors are located on a regular grid of 5 pixels following the usual practice and considered a uniform patch size of 60 × 60 pixels as its support on the basis of an empirical study. This SIFT descriptor at each location (grid point) is fed as a 128-dimensional input feature vector to a Multilayer Perceptron (MLP) network which gives response for each grid point as either text or non-text. Depending on certain aggregate response at each pixel we localize text regions in the input video frame. Next, we employ K-means clustering to detect the text components present in the localized region of the video frame. Finally, two simple rules are applied to decide certain possible detected text components as noise. We obtained encouraging simulation results of this approach on a variety of video lecture frames.**

Keywords—Reading of video camera based white or color board notes; SIFT descriptor; MLP network; K-means algorithm; Hand-written text localization;

I. INTRODUCTION

Although much progress in the field of handwriting recognition has been achieved, the existing methods for such tasks perform efficiently when the input data is somewhat well behaved. But in real life situations, we often encounter handwritten data which are not so well behaved. For example, when a teacher teaches in a class, he or she writes on the board (not necessarily a whiteboard) in an unconstrained manner. An automated system may take video of such classroom teachings which may be processed later to extract the text portions in order to convert the class notes into an editable text. Such a problem belongs to a new application of handwriting analysis tasks compared to the classical handwriting recognition applications like automatic processing of bank cheques [1] or postal documents to more challenging problems like recognition of historical manuscripts [2], personal memos or interpretation of hand drawn sketches [3] and lately to recognition of unconstrained whiteboard notes [4, 5].

Execution of collaborative works (e.g. brainstorming, discussions, and presentations) is quite common in both corporate and academic environments. In such environments, lectures are organized on a regular basis. Although in the corporate environments, whiteboards are commonly used but in classroom teachings academicians usually prefer traditional non-whiteboards. In the literature, there exists only a few works [4, 5] towards recognition of texts from the image of a whiteboard used by the presenter in a smart environment. To provide not just a digital capture of the whiteboard, but also the recognition of that content in an interactive software framework, is one of the final goals of such a smart lecture room. Systems used by such a smart lecture theatre include certain specific (often costly) hardware e.g., special whiteboard, multiple cameras, pen, wireless microphone proposed by the e-Learning system [6] etc.

In contrast to the above scenario of a smart lecture theatre, we propose a novel text detection methodology from video frames of classroom lectures using any possible type of board (white, black or any other color) towards the ultimate goal of the recognition of handwritten texts on them. The realization of such a handwritten text recognition system would definitely help the teachers and students of Colleges and/or Universities to prepare editable notes from the videos of lectures or realization of mind map [7] etc. A sample video frame of lecture used in the present study is shown in Fig. 1.

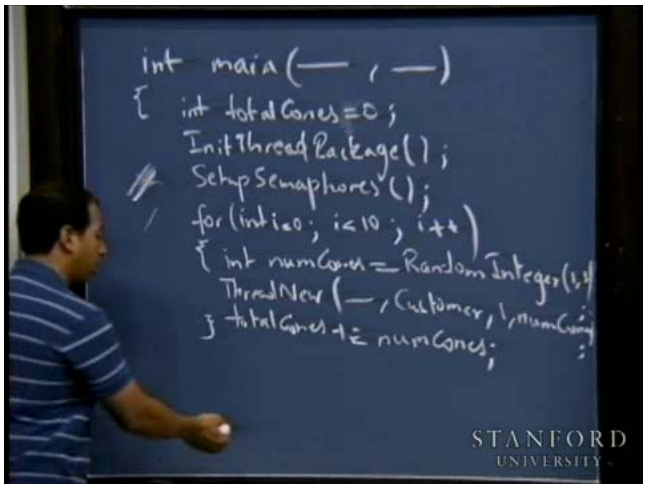


Fig. 1. A sample of video lecture frame used in the present study.

2167-6445/14 $31.00 © 2014 IEEE
DOI 10.1109/ICFHR.2014.110

The current system focuses on localization of the handwritten texts on white / colored boards which could later be fed to a handwriting recognition system. The goal of such an attempt is to obtain a digital, editable note of the lecture and also the achievement of possible interactions with the static lecture content on such a board.

The remainder of this article is organized as follows. Section II discusses existing related studies dealing with the recognition of texts on white / colored board. Section III describes the details of the proposed approach for localization and detection of texts from video lectures. The experimental setup including the sample data used for training and testing of the proposed approach has been detailed in Section IV. Finally, Section V summarizes the present study highlighting the strengths and weaknesses of the proposed approach.

II. BACKGROUND

Over the last two decades impressive progress could be achieved in the field of handwriting recognition, even for large vocabularies [8] and multi-writer scenarios. However, a crucial issue has not yet been completely solved: well-behaved data with some reasonable distinction between foreground and background pixels is necessary to obtain efficient offline recognition results. However, such conditions may not hold valid in the white / colored board reading scenario addressed in this article. In such a scenario, the presenter usually writes texts quite casually on the board and the video camera captures image of the writing board along with certain other objects possibly including the presenter himself/herself.

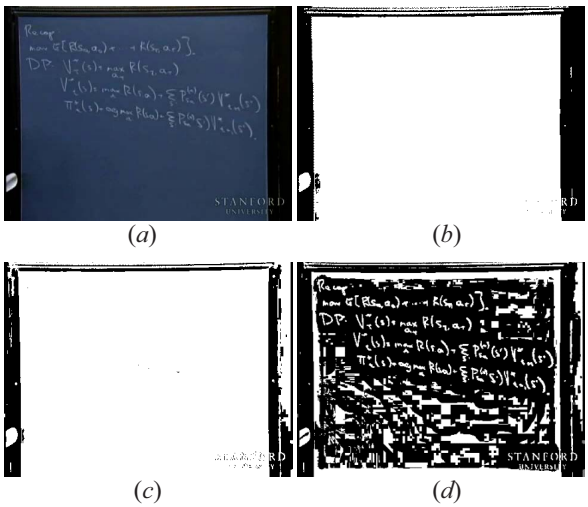
As an alternative to the systems based on electronic whiteboards for smart meeting rooms, Wienecke et al. [9] proposed a prototype system for automatic reading of video-based unconstrained handwritten texts on whiteboard. Although this system involving an incremental processing strategy in which text lines were recognized as soon as they were visible in the video image could be effective as an alternative to electronic whiteboard system, it could handle the simplistic situations consisting of only continuous texts.

Recognition of text written on an interactive whiteboard (based on eBeam technology of Luidia Inc.) was also studied in [10], [11]. In these studies, an off-the-shelf offline handwriting recognizer based on an HMM (hidden Markov model) was used although the input data provided by the sensing device was in the on-line format. Initially [10], the authors achieved a recognition rate of 64.3% and later [11] they improved this performance to 68.5%. In both of these two studies, the authors considered the situation when only continuous texts were written on the whiteboard.

On the other hand, nearly a decade back recognition of keywords written on whiteboard was studied [6] towards the development of an e-learning system. This system considered a complex hardware scenario including two cameras controlled by a computer, a special pen capture tool and a microphone. Handwritten character strings were segmented from the whiteboard image provided by the pen capture tool. This early study showed that the recognition accuracies of such handwritten characters are not enough for recognition of the keywords.

Recently, the layout analysis of realistic whiteboard notes had been studied in [4], [5] and [12]. In [5], the authors presented a three-stage approach for automatic layout analysis of camera-based whiteboard documents. Here, the authors assumed a reasonable foreground-background separation of the handwriting and a locally adaptive binarization followed by connected component extraction were applied obtaining handwritten elements which were next automatically classified as either simple graphical elements of a 'mind map' or elementary text patches. Finally, a clustering technique was used generating hypotheses for those image regions where textual annotations of the 'mind map' could be found. In these studies, extraction of connected components was based on binarization of whiteboard image by Sauvola's thresholding technique [14] which is an optimized version of the basic locally adaptive binarization method due to Niblack [15]. These authors also analyzed an efficient implementation [16] of Sauvola's binarization approach.

However, the above binarization strategy often fails to separate foreground-background for the present data of video lecture frames. In fact, in the present study, we empirically analyzed standard global and local thresholding approaches on our image samples and these binarized outputs corresponding to one such sample image are shown in Fig. 2. It may be seen that, after binarization using the global method due to Otsu [13], no text components of the input image (shown in Fig. 2(*a*)) are found in its binarized image (Fig. 2(*b*)). Also, the binarized output (shown in Fig. 2(*c*)) of the same input sample obtained by the locally adaptive method due to Sauvola [14] using 15 × 15 window size has the same fate with no handwritten text components of the input image retained. However, when this image is binarized using the basic local method due to Niblack [15] and 15× 15 window size, the text components appear in the binarized output along with significant proportion of noise components as shown in Fig. 2(*d*). It is not a trivial task to filter out such noise components retaining only the text elements of this binarized image.



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

Fig. 2. Output of different binarization techniques on input image (*a*). Here, (*b*), (*c*) and (*d*) are the binarized output of Otsu, Sauvola and Niblack binarization techniques, respectively.

The difficulties of localization of handwritten components in video frames of lectures are discussed above. The various existing strategies of locating texts in usual video frames such as the one described in [17-19], are also not suitable for the present problem since the handwritten components of present sample data often include various annotation marks, complex mathematical formulas in addition to their non-aligned nature, and several other peculiarities. We describe below our recent study of a novel strategy towards a solution to this problem.

III. Proposed Methodology

In this Section, we provide the details of the proposed methodology for detection of handwritten texts on a board used in lecture halls. In the preprocessing stage, we convert the input color image of a video frame to its grayscale image. Here, we do not consider any other preprocessing operation and directly move to the feature extraction stage where we densely extract SIFT descriptors [20] (128 dimension) which are used as inputs to a Multilayer Perceptron (MLP) locating text blocks in a video frame. The classification results provided by the MLP are used to localize texts in a video frame. Next, the well-known K-means algorithm is used to detect text components in the localized regions. Finally, a few non-text components which may be detected by our text detection approach are removed following two simple rules. Further details are provided below.

A. Dense Feature Extraction

We densely extract SIFT descriptors on the input video frame. These descriptors, located on a regular grid of 5 × 5 pixels, have a uniform scale of 60 × 60 pixels. Thus each window of 60 × 60 pixels has approximately 92% overlap with each of its adjacent windows. The above size of the regular grid is considered based on the usual practices [21] and the value of the uniform scale has been empirically decided based on the present database. Fig. 3(*a*) and Fig. 3(*d*) respectively show a video frame and its dense grid of SIFT descriptors.

Here, we considered the same idea of calculation of the dense descriptors to a number of separable convolutions as it had been done previously in [21]. The SIFT descriptor at a grid point (x_0, y_0) on the input video frame I is a histogram of the gradient $\nabla I(x,y)$ in a circular patch surrounding that point [22]. The histogram is indexed by the relative position $(x - x_0, y - y_0)$ and orientation $\angle\nabla I(x,y)$ of the gradient $\nabla I(x,y)$ in the patch, weighed by the gradient modulus $|\nabla I(x,y)|$ and by a Gaussian window centered at (x_0, y_0). The relative positions are quantized into 4 × 4 bins and the orientation in 8 bins using bilinear interpolation. For a given orientation, the data for a bin b is obtained by computing integrals of the type $\int g(x - x_0, y - y_0) h_b(x - x_0, y - y_0) f(x,y) dxdy$, where $f(x,y)$ is the mass of the gradient at that particular orientation, $g(x,y)$ is the Gaussian window and $h_b(x,y)$ is the product of two triangular windows resulting from the bilinear interpolation of bin b.

It may be noted that the computation of the dense descriptors at a location requires a total number of 4 × 4 × 8 separable convolutions. Since if the Gaussian window $g(x,y)$ be dropped, the effect on the computed descriptors is modest and convolutions for different spatial bins at the same orientations are identical up to translation, and only 8 separable convolutions are sufficient [22], we followed the same in the present study and used the open source implementation of [23] (VLFeat 0.9.16) to compute a histogram of feature occurrences within a window of 60 × 60 pixels using integral image for each pixel on a grid with a step of 5 pixels. Two intermediate stages of this computation approach of the SIFT descriptor are explained in Fig. 3(*b*) and Fig. 3(*c*).

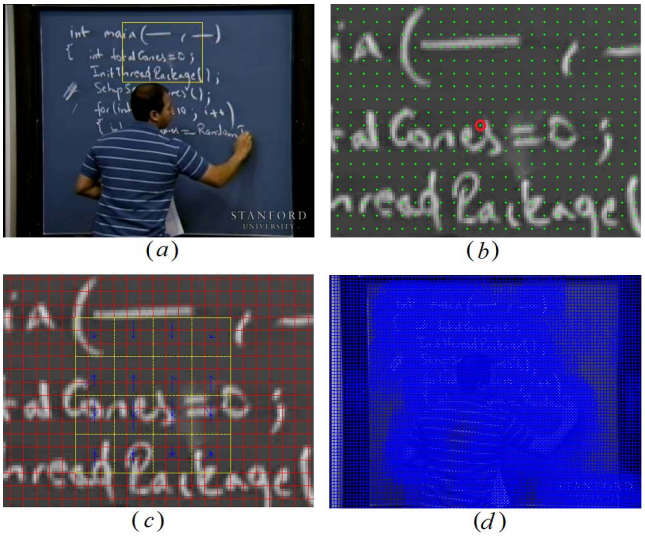


Fig. 3. An illustration of SIFT descriptor computation: (*a*) An input video frame. A small part of it is marked by yellow rectangle. (*b*) The marked region of Fig. 3(*a*) is magnified and the green dots are the regularly spaced grid points on it at an interval of 5 pixels. One such grid point is marked by red circle. (*c*) A window of 60×60 pixels around the marked grid point of Fig. 3(*b*) is shown by a rectangle of continuous yellow boundary. A division of this window into 4×4 smaller windows is shown by dotted yellow lines. Each of these smaller windows consists of 3×3 grid corresponding to Fig. 3(*b*) and this grid is shown by red color. The SIFT descriptor consisting of gradient values computed over each of the 4×4 windows and quantized into 8 bins are shown by blue color arrows at respective center grid points. Lengths of arrows at each location correspond to the values at respective bins. (*d*) The dense SIFT descriptors for the whole frame of Fig. 3(*a*) as provided by the implementation VLFeat 0.9.16.

B. Classification and Localization of Text Regions

The above SIFT descriptor at each location (grid point) is fed as a 4 × 4 × 8 = 128-dimensional input feature vector to a Multilayer Perceptron (MLP) network which gives response for each grid point as either text or non-text. The grid points of the input video frame (shown in Fig. 4(*a*)) which are recognized as text by a trained MLP network have been shown in Fig. 4(*b*) along with the respective strengths of the SIFT descriptor at such locations. Thus, the MLP network recognizes each 60×60 input block of video frame as 'text' (class 1) or 'non-text' (class 0). Since text blocks are selected in an overlapping manner, a pixel in it belongs to multiple text blocks. Thus, we maintain a score value (initialized with the value 0) for each pixel and whenever a text block is recognized by the MLP as 'text' (class 1) or 'non-text' (class 0), its score value gets added with the numerical value of the class identifier (1 or 0). Since these blocks are overlapping at a jump of 5 pixels both in the horizontal and the vertical directions, each pixel of the video frame belongs to (60÷5) × (60÷5) = 144 blocks (boundary situations are taken care of following a standard practice). Thus, if each of these 144 blocks to which

certain pixel belongs to are identified as text blocks by the MLP network, then its final score value is 144. Similarly, if each of these 144 blocks to which certain pixel belongs to are identified as non-text blocks by the MLP network, then the corresponding final score value is 0. In fact, possible final score value of any pixel lies in the range 0 to 144.

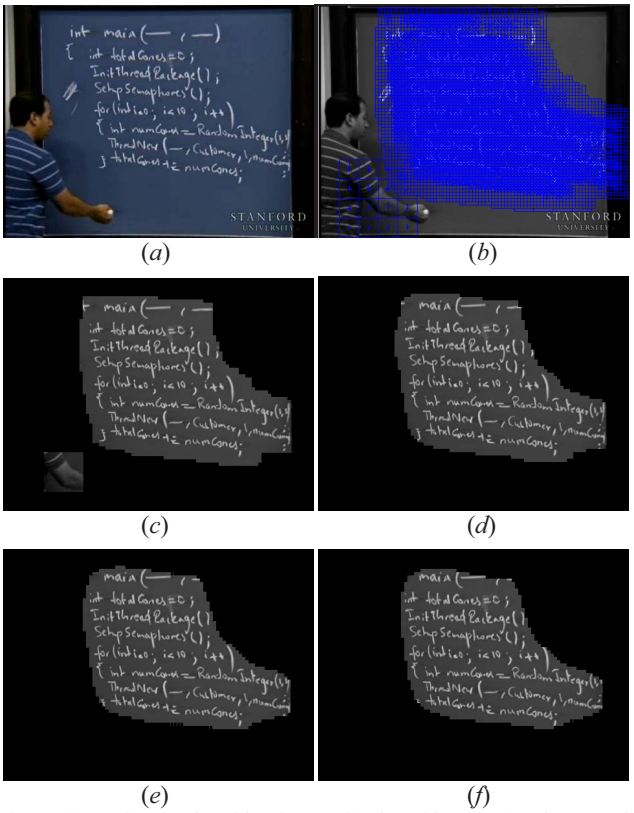


Fig. 4. (*a*) An input color video frame. (*b*) The grid points (on the grayscle imge of the video frame of (*a*)) of the dense SIFT descriptor which have been recognized by an MLP network as 'text' are shown in blue color along with the respective strengths of the SIFT descriptor.The black pixels of (*c*), (*d*), (*e*) and (*f*) are the pixels for each of which the final score value is $\leq \rho$. The value of ρ is 0, 5, 10 and 15 for output images of (*c*), (*d*), (*e*) and (*f*) respectively.

We introduce here the concept of using a threshold value on the above final score. If this final score is more than this threshold ρ, then the corresponding pixel is identified as a text pixel. In the present study, we considered a number of values of ρ and the text localization results using ρ = 0, 5, 10 and 15 are shown in Figs. 4(*c*), (*d*), (*e*) and (*f*) respectively. On the basis of the sample database used in the present study, we select ρ = 5 for obtaining results of the present module localizing texts in the input video frame.

C. Text Component Detection

After localization of text regions we detect individual text components present in this region. To achieve the same, we apply the K-means (K=2) algorithm on the pixel gray values of this detected region. The cluster containing larger number of pixels corresponds to background pixels and these pixels are turned 'black' and the pixels of the other cluster correspond to text pixels which are turned 'white'. In Figs. 5 (*a*) and 5(*c*) localized texts are shown for two video frames and their detected text components are shown in Figs. 5(*b*) and 5(*d*) respectively.

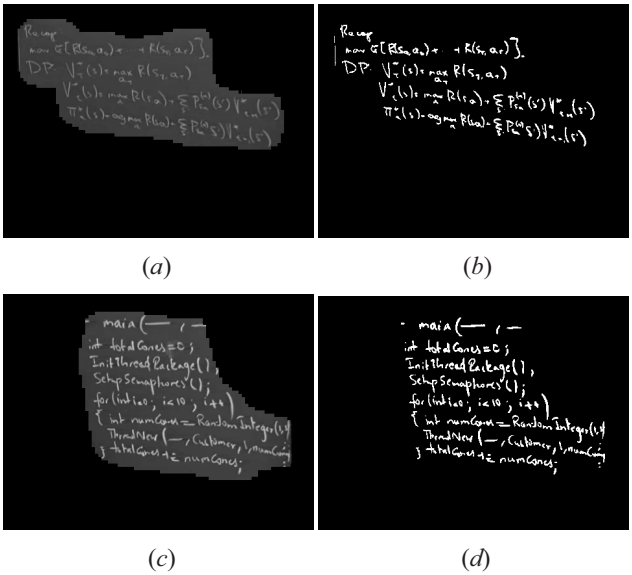


Fig. 5. (*a*), (*c*) Localized texts from two different video frames. (*b*), (*d*) Text components detected by using K-means (K=2) clustering algorithm on the text localized image shown in (*a*) and (*c*) respectively.

IV. Experimental Results

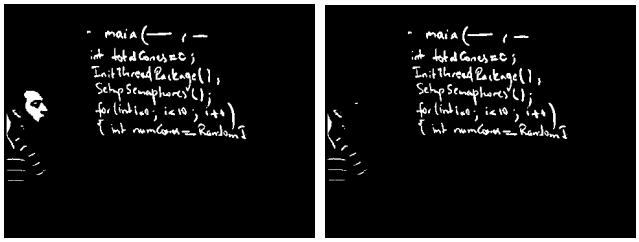
The dataset used in our experimentation consists of video frames of lectures delivered by 5 different speakers. For each lecture, we considered 20 frames at a fixed gap of 100 frames. Among these 100 frames, there are 6 frames which do not have any text. However, we did not exclude them from our sample frame database. We randomly selected 30 frames from this database to form the test set and another 20 frames to form the validation set. The remaining 50 frames had been used to form the training samples for training of the MLP network. The MLP used in the present study consists of 128 input nodes, 15 hidden nodes in a single hidden layer and 2 output nodes. Its initial training iteration using the well-known back-propagation algorithm was continued till the recognition performance on a set of validation samples continued to improve

Generation of training samples for training the MLP network is a tricky job. Although the text portions may not have much variation, the non-text regions usually show significant variations such as blank portions of the board, frame of the board, duster, table top, floor, roof, speaker and so on. This scenario is actually noted when we feed the validation samples as input of the MLP after its initial training with the frames in the training set. A significant proportion of validation samples from different parts of the frame which includes various other objects (belonging to the non-text class) get misclassified as text. We have added these misclassified validation samples to the original set of training samples to continue further training of the MLP network for a while (we considered 100 iterations for this second round of training of our experiment). The MLP network, after its second round of training, was used for text localization on the test frames.

Fig. 6. Text extraction results on 3 samples of the present test database are shown. (*a*), (*b*), (*c*) are three test samples. (*d*), (*e*), (*f*) shows respective sets of grid points identified by the MLP as belonging to 'text' category. (*g*), (*h*), (*i*) are the respective localized texts. (*j*), (*k*), (*l*) are the three final outputs of the proposed text detection approach.

For demonstration of the output at various stages of the proposed approach, we consider three video frames shown in Figs. 6(*a*), 6(*b*) and 6(*c*). The grid point classification results of the MLP on these frames are shown in Figs. 6(*d*), 6(*e*) and 6(*f*) and the respective localized text regions are shown in Figs. 6(*g*), 6(*h*) and 6(*i*). Finally, the detected text components from respective video frames are shown in Figs. 6(*j*), 6(*k*) and 6(*l*).

Since we are not aware of the availability of any standard database of similar video frames along with the ground truths of the text and non-text components, obtaining of numerical evaluation results for the proposed approach is a difficult task. However, for the 30 frames used as the test samples in our experiment, we manually evaluated the text detection results and observed that approximately 89% of the text components in these video frames are correctly extracted by the proposed method. The low detection rate of the proposed method is due to several factors such as the poor resolution, low contrast, very small text components compared to the whole frame etc. On the other hand, false detection of non-text components is very low. There are only three samples in our test database for each of which a few non-text components have been detected as text components and one of these output images is shown in Fig. 7(*a*). A few of these misclassified components can be identified with the help of a few simple rules. In Fig. 7(*b*), the non-text component corresponding to the head of the presenter is removed with the help of the following two rules: a component with (i) pixel density greater than 90% and (ii) volume greater than 250 cannot be a text component. It may be noted here that these simple rules are unable to remove noise components originated from the stripes in the shirt of the presenter.



(*a*) (*b*)

Fig. 7. (*a*) Text dectection result of the proposed method on a video frame of our test database. (*b*) after removal of a few noise components from the output shown in (*a*) by appling certain simple rules.

V. SUMMARY

In this article, we presented a novel work for automatic detection of text components from video frames of class room lectures. Although there exists a few studies for detection of texts on whiteboards from video frames, to the best of our knowledge, there does not exist any work in which localization of texts on color (black, green, blue etc.) boards had been studied. In the above sense, the present study has a novelty. On the other hand, to the best of our knowledge, the use of SIFT features along with MLP network for text extraction purpose had not been reported earlier. The video frames used in the present study had been collected from video lectures available online and thus the results of our experimentation reflect the robustness of the proposed method. Also, the proposed method has been found to detect text components on whiteboards equally efficiently.

Although there exists several studies [24, 25] in the literature for text extraction from natural scene images, those fail to extract handwritten texts from images of video frames of lectures due to the specific assumptions used in these studies.

In future studies, we aim to design an efficient preprocessing module which should improve the localization results to a great extent. Moreover, we aim to develop a ground truth dataset for such video frames to pave the path for more effective studies on this problem area.

ACKNOWLEDGMENT

This research has been partially supported by the Indian Statistical Institute, Kolkata and the Society for Natural Language Technology Research, Kolkata.

REFERENCES

[1] R. Palacios, A. Gupta, "A system for processing handwritten bank checks automatically," Image and Vision Computing, vol. 26, no. 10, pp. 1297–1313, 2008.

[2] V. Romero, N. Serrano, A. H. Toselli, J. A. Sanchez and E. Vidal, "Handwritten Text Recognition for Historical Documents," Proc. of Language Technologies for Digital Humanities and Cultural Heritage Workshop, pp. 90–96, Hissar, Bulgaria, 16 September 2011.

[3] S. Ahmed, M. Weber, M. Liwicki, C. Langenhan, A. Dengel, and F. Petzold,"Automatic analysis and sketch-based retrieval of architectural floor plans," Pattern Recognition Letters, vol. 35, pp. 91-100, January 2014.

[4] T. Plotz, C. Thurau and G. A. Fink, "Camera-based whiteboard reading: New approaches to a challenging task," Proc. Int. Conf. on Fontiers in Handwriting Recognition, pp. 385-390, 2008.

[5] S. Vajda, T. Plotz, and G. A. Fink, "Layout analysis for camera-based whiteboard notes," Journal of Universal Computer Science, vol. 15, no. 18, pp. 3307-3324, 2009.

[6] D. Yoshida, S. Tsuruoka, H. Kawanaka and T. Shinogi, "Keywords recognition of handwritten character string on whiteboard using word dictionary for e-learning," Proc. of Int. Conf. on Hybrid Information Tech. (ICHIT 2006)– vol. 1, Washington, pp. 140-145, 2006.

[7] P. Farrand, F. Hussain and E. Hennessy, "The efficacy of the 'mind map' study technique," Journal of Medical Education, vol. 36, no. 5, pp. 426-431, 2002.

[8] A. L. Koerich, R. Sabourin, and C. Y. Suen,"Large vocabulary off–line handwriting recognition: A survey," Pattern Analysis and Applications, vol. 6, no. 2, pp. 97-121, 2003.

[9] M. Wienecke, G. A. Fink and G. Sagerer, "Toward automatic video-based whiteboard reading," Int. Journal on Document Analysis and Recognition, vol. 7, no. 2, pp. 188–200, 2005.

[10] M. Liwicki and H. Bunke, "Handwriting recognition of whiteboard notes," Conference of the International Graphonomics Society, pp. 118-122, 2005.

[11] M. Liwicki and H. Bunke,"Handwriting recognition of whiteboard notes - studying the influence of training set size and type," International Journal of Pattern Recognition and Artificial Intelligence, vol. 21, no. 1, pp. 83-98, 2007.

[12] S. Vajda, L. Rothacker, G. A. Fink,"A Method for Camera-Based Interactive Whiteboard Reading," Proc. Int. Workshop on Camera-Based Doc. Anal. and Recog., LNCS 7139, Springer-Verlag, pp. 112-125, 2011.

[13] N. Otsu, "A Threshold Selection Method from Gray-Level Histograms," in IEEE Trans. on Systems, Man and Cybernetics, vol. 9, issue 1, 1979.

[14] J. Sauvola and M. Pietikäinen, "Adaptive document image binarization," Int. Journal of Pattern Recognition, vol. 33, pp. 225-236, 2000.

[15] W. Niblack, An introduction to digital image processing. Birkeroed, Denmark: Strandberg Publishing Company, 1985.

[16] F. Shafait, D. Keysers, T. M. Breuel, "Efficient implementation of local adaptive thresholding technique using integral images," Proc. Document Recognition and Retrieval, pp. 101–105, California, USA, 2008.

[17] S. Bhowmick and P. Banerjee, "Bangla Text Recognition from Video Sequence: A New Focus", Proc. NaCCS, pp. 62-67, 2012.

[18] P. Banerjee and B. B. Chaudhuri, "An Approach for Bangla and Devanagari Video Text Recognition," Proc. International Workshop on Multilingual OCR (MOCR), Washington DC, USA, article no. 8, 2013.

[19] P. Banerjee and B. B. Chaudhuri, "Video text localization using wavelet and shearlet transforms," Proc. SPIE 9021, Document Recognition and Retrieval XXI, 90210B (December 27, 2013); doi:10.1117/12.2036077.

[20] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," Int. Journ. of Comp. Vision, vol. 60, no. 2, pp. 91-110, 2004.

[21] L. Rothacker, G. A. Fink, P. Banerjee, U. Bhattacharya and B. B. Chaudhuri, "Bag-of-Features HMMs for segmentation-free Bangla word spotting," Proc. Int. Workshop on Multilingual OCR (MOCR), Washington DC, USA, article no. 5, 2013.

[22] B. Fulkerson, A. Vedaldi and S. Soatto, "Localizing objects with smart dictionaries," Lecture Notes in Computer Science, vol. 5302, pp. 179-192, 2008.

[23] A. Vedaldi, B. Fulkerson, Vlfeat: Feature extraction library (2012), http://www.vlfeat.org/.

[24] A. Roy Chowdhury, U. Bhattacharya and S. K. Parui, "Scene text detection using sparse stroke information and MLP", Proc. of Int. Conf. on Pattern Recog., pp. 294-297, IEEE Comp. Soc. Press, 2012.

[25] S. Banerjee, K. Mullick and U. Bhattacharya, "A robust approach to extraction of texts from camera captured images", Proc. of the 5th Int. Workshop on Camera-Based Doc. Anal. and Recog. (CBDAR 2013), held at Washington DC, USA on 23 Aug., 2013, pp. 53-58, 2013.