TEXT DETECTION AND RECOGNITION IN NATURAL SCENES AND CONSUMER VIDEOS

Arpit Jain‡, Xujun Peng, Xiaodan Zhuang, Pradeep Natarajan, Huaigu Cao

Speech, Language and Multimedia Business Unit
Raytheon BBN Technologies, 10 Moulton Street, Cambridge, MA 02138
{ajain,xpeng,xzhuang,pradeepn,hcao}@bbn.com

ABSTRACT

We propose an end-to-end system for text detection and recognition in natural scenes and consumer videos. Maximally Stable Extremal Regions which are robust to illumination and viewpoint variations are selected as text candidates. Rich shape descriptors such as Histogram of Oriented Gradients, Gabor filter, corners and geometrical features are used to represent the candidates and classified using a support vector machine. Positively labeled candidates serve as anchor regions for word formation. We then group candidate regions based on geometric and color properties to form word boundaries. To speed up the system for practical applications, we use Partial Least Squares approach for dimensionality reduction. The detected words are binarized, filtered and passed to a hidden Markov model based Optical Character Recognition (OCR) system for recognition. We show significant improvement in text detection and recognition tasks over previous approaches on a large consumer video dataset. Furthermore, the event detection system built upon the OCR output of this approach outperformed multiple other OCR-only based submissions in the recently concluded NIST TRECVID 2013 multimedia event detection evaluations.

Index Terms— text detection and recognition, consumer video, event detection, Partial Least Squares

1. INTRODUCTION

An end-to-end system for text detection and recognition is important in multiple domains such as content based retrieval systems, video event detection, human computer interaction, autonomous robot or vehicle navigation and vehicle license plate recognition. There are several commercial systems for text recognition in scanned document[1][2]. However, these systems typically need cropped and binarized text regions to perform well for natural scene text[3]. Text detection in natural scenes is a challenging problem and has gained a lot of attention recently [4]. Such texts presents low contrast with background, large variation in font, color, scale and orientation combined with background clutter. Therefore a robust and fast recognition system is desirable.

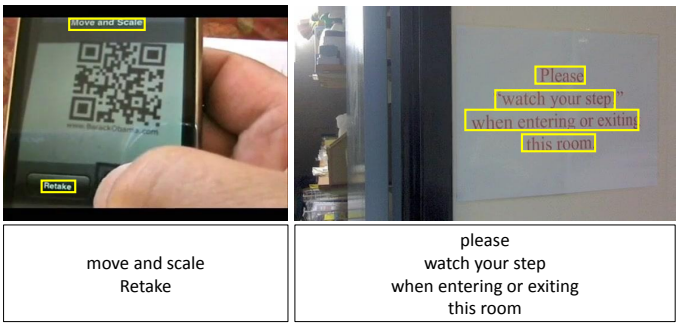


Fig. 1. Proposed end-to-end system for text detection and recognition

Text detection approaches can be divided into two main categories (a) sliding window based approaches (b) connected components based approaches. In sliding window based approaches, low-level features are extracted for each scanning window and each candidate is evaluated for presence of text using machine learning techniques. [5] used Histogram of Oriented Gradient (HOG) features together with Random fern technique for text recognition. The ambiguities in recognition were fixed using a pictorial structure with a lexicon for text detection and recognition. [6] used gradient, edge, texture and Gabor features together with adaboost learning technique for classification. [7] classified text windows from non-text using principle stroke Gabor words and showed improvement over previous approaches. However, these approaches do not explicitly account for scale variations and therefore most of these approaches are applied over multiple scales and results are aggregated into single detection result.

On the other hand, connected component based approaches first extract pixel regions which have similar edge strength, color, texture or stroke width and evaluate each one of them for being text or not text using rule-based or machine learning techniques. [8] used low-variance in text stroke width as a measure to select text candidate regions from non-text regions. [9] extracted candidate regions using edgelinks (continuous edge chains) and evaluated each candidate using a Support Vector Machine (SVM) classifier. The output of the SVM is integrated using a Conditional Random Field (CRF). Recently, Stable Extremal Regions has become

‡ Arpit Jain is currently a graduate student at University of Maryland College Park in Electrical and Computer Engineering Department. He can be reached at arpitjain1@gmail.com

978-1-4799-2893-4/14/$31.00 ©2014 IEEE

popular approach to extract connected component candidates [10] as they are robust to illumination and scale changes. [11] showed a real-time system for text detection in videos using extremal regions. [12] proposed geometric grouping over MSER regions and classified the regions using adaboost. [13] built a graph network over MSER candidates and determined text from non-text region using graph cut.

We propose an end-to-end system for video text detection and recognition. The proposed system comprises of three steps (a) text localization (b) text line aggregation (c) text line recognition. We use MSER regions as candidates and instead of using rules or geometric based grouping, we apply a text/non-text SVM classifier over each candidate. We compute rich shape descriptors and compresses them to very few dimensions while preserving discriminability using Partial Least Squares (PLS) technique. PLS technique enables use of a large set of features in classification, and speeds up the classification significantly. Each positively labeled candidate serves as an anchor region around which we group candidate regions based on geometric and color properties. At this step, we allow negatively labeled candidates to take part in text line aggregation, to overcome mistakes in the classification step. We binarize the detected text regions and pass it to an OCR system for word recognition.

2. TEXT LOCALIZATION

2.1. Text Candidates using MSER

MSER technique, proposed by Matas et. al.[10], finds stable connected regions over a range of thresholds. This technique was originally used for correspondences between two images with different viewpoints. Low-level image segmentation as a prerequisite step of text detection can also benefit from MSER. MSER is able to detect most of the characters even in low resolution video frames. We prune candidates of sizes smaller than a predefined threshold t_l or larger than t_h. We also prune candidates of aspect ratios outside the range [r_l,r_h], and with numbers of holes beyond a threshold h_{th}. After MSER candidates are extracted, we compute features for training a text/non-text SVM classifier.

2.2. Feature Extraction

We extract three types of features for classifying candidate regions into text or non-text (background).

Histogram of Oriented Gradients (HOG), proposed by [14], showed impressive result for object and human detection. With an image divided into cells, HOG features are rich shape descriptors which captures the shape of the object by quantizing the gradient information in each cell. These cells are grouped into equal or larger sized overlapping blocks which are then normalized and concatenated together to form the feature vector. Figure 2 shows visualization of HOG features for letter A and X. Each cell is represented by an oriented "star" showing the strength of corresponding gradient direction. In order to keep the feature dimensions consistent, all the text candidate regions are resized to fixed size before computing HOG features.

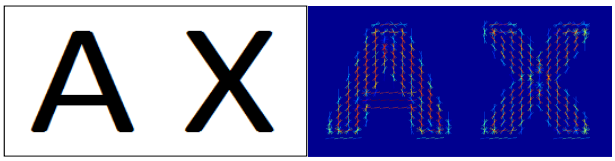


Fig. 2. Visualization of HOG features

Gabor filter is a band-pass filter which can be viewed as a sinusoidal plane of particular frequency and orientation modulated by a Gaussian function. It extracts orientation-dependent frequency information such as direction of strokes which can be used to discriminate text from non-text. We use the standard deviation of output of Gabor filters on candidate regions as feature. The 2-D Gabor filter can be written in the following form:

$$G(x, y, \lambda, \phi, \sigma_x, \sigma_y) = \frac{1}{2\pi\sigma_x\sigma_y} \exp\left\{\frac{1}{2}\left(\frac{R_1^2}{\sigma_x^2} + \frac{R_2^2}{\sigma_y^2}\right)\right\} \exp\left\{i\frac{2\pi R_1}{\lambda}\right\}$$

where $R_1 = xcos\phi + ysin\phi$ and $R_2 = ycos\phi - xsin\phi$, λ is the wavelength of Gabor filter, ϕ is the orientation of Gabor filter and σ_x and σ_y denotes the standard deviation of Gabor filter. For simplicity, $\sigma_x = \sigma_y = \sigma$.

We compute the ratio of corners to edges [9] for each text region according to the following equation,

$$\xi = \sum_{x=1}^{w}\sum_{y=1}^{h} C(x, y) / \sum_{x=1}^{w}\sum_{y=1}^{h} E(x, y), \quad (1)$$

where $w \times h$ is the size of bounding box for the text region, $C(x, y)$ denotes the intensity of corner obtained after binarization with a fixed threshold of Harris corner detection [15] result over the frame, $E(x, y)$ denotes the intensity of edge map of the input image obtained using Canny Edge detection algorithm [16]. We also use the x, y coordinates, width and height of the boxes as additional features.

Our preliminary experiments show that even simple concatenation of the three types of features outperforms any single type on detection accuracy.

2.3. Dimensionality reduction using PLS

Speed is an important factor when we are building a practical system for text detection and recognition in videos. We found that SVM classification on original features is the bottleneck in computational efficiency. Hence, we apply PLS technique for dimensionality reduction, compressing the original feature space (2066 dimensions) to just few dimensions (9 dimensions) without reducing detection accuracy. This gives us 5x speed up, which is significant given the size of our dataset. We briefly describe mathematical formulation of PLS technique below. More detailed discussion can be found in [17].

Let $X_{n\times m} \subset \mathbb{R}^m$ denote an m dimensional feature vectors of sample size n and let $Y_{n\times 1} \subset \mathbb{R}$ be their corresponding 1-dimensional class labels. PLS decomposes the zero-mean matrix $X_{n\times m}$ and zero-mean $Y_{n\times 1}$ into

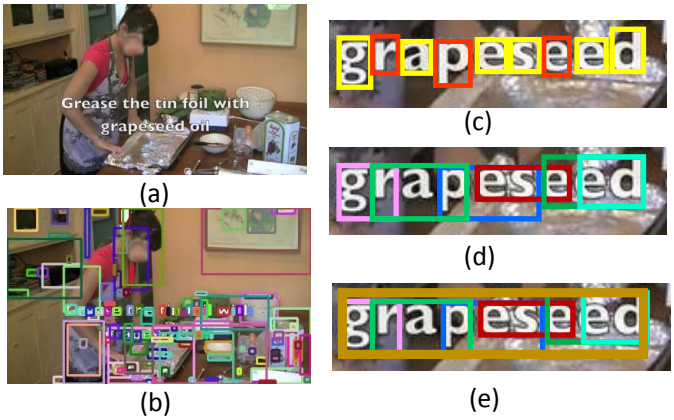


Fig. 3. (a) original image, (b) MSER candidates, (c) SVM classifier result (positive in yellow and negative in red), (d) *grouplets* after merging (each *grouplet* showed in different color), (e) detected text bounding box

$$X = TP^T + E$$
$$Y = Uq^T + f \quad (2)$$

where T and U are $n \times p$ matrices containing p extracted latent vectors. The matrix $P_{m\times p}$ and $q_{1\times p}$ represents the loading, similar to Principal Component Analysis (PCA). E and f represents residual error while projecting data onto lower subspace for X and Y respectively. The Nonlinear iterative partial least squares algorithm (NIPALS) [17] constructs a set of weight vectors $W = w_1, ..., w_p$ such that,

$$[cov(t_i, u_i)]^2 = \max_{|w_i=1|} [cov(Xw_i, y)]^2 \quad (3)$$

where t_i and u_i are the i-th columns of T matrix and U matrix respectively, and $cov(t_i, u_i)$ is the covariance between latent vector t_i and u_i.

PLS find subspaces where the covariance between projected feature X and label Y is maximized. A key difference between PCA and PLS is that PLS exploits label information while finding latent subspace. The resultant W matrix is used to project data into the low dimensional subspace which is used to learn SVM classifier on training data and classify text from non-text regions during testing.

2.4. SVM classifier

We extract MSER regions from training data, which are separated into positive and negative data according to manual annotation of text bounding boxes. A region is considered positive only if it overlaps more than 90% with a ground truth bounding box. We extract features from each region and project them onto a lower PLS subspace. We then learn a SVM classifier in the projected subspace [18].

3. GROUPING OF LOCALIZED TEXT REGIONS

Each positively classified MSER region serves as an anchor for grouping the text into words during testing. We want to emphasize at this point that the previous classification step is only used to localize the regions with high text probabilities. MSER regions misclassified by the SVM classifier will still be considered for potential merging with these anchors if they satisfy certain criteria. This procedure allow us to overcome the mistakes of the classification step when grouping text regions into words.

For each positively classified MSER region, we search its neighborhood for MSER regions which have similar color, size, aspect ratio and which satisfy proximity criteria to form a word. At this step, we consider all the initial candidates irrespective of their classification label. If a MSER region satisfy the criterions for merging, then the anchor and the searched regions are merged into a '*grouplet*'. Each positive anchor can at most connect to two adjacent regions and a single region can be part of multiple *grouplets* (Figure3(d)). If an anchor does not connect to any neighboring region, then it is discarded. All the regions which do not merge are also discarded from further analysis.

We then follow a simple heuristic scheme to merge these *grouplets* into words. Two *grouplets* will be merged if they are spatially close and if they have similar color, height and aspect ratio. This step is continued until no other *grouplet* can be merged with one another. The bounding boxes obtained after grouplet merging are considered the final detection results.

Figure3 shows the steps of text grouping. Though the letters "r", "p" and "e" are misclassified as non-text by SVM, they get merged into different *grouplets*.

4. OCR DECODING

We pre-process each text line before OCR decoding. Each cropped textline image is first binarized using Otsu method [19]. If the text is light on dark background, textline image is inverted before the thresholding is applied. A median filter is applied to remove salt and pepper noise. Finally, the textline image is resized to a fixed height of 110 pixels with its aspect ratio unchanged and passed for OCR decoding.

4.1. OCR system

We use the BBN HMM Byblos OCR system for decoding [20]. We briefly describe the mathematical model used in OCR system as follows. Lets assume that textline is represented by a sequence of feature vectors X. The goal is to find sequence of characters (C) that best explain the features X. Mathematically this can be written as $P(C|X)$ which when expanded using Bayes' rule,

$$P(C|X) = \frac{P(X|C)P(C)}{P(X)}, \quad (4)$$

where $P(X|C)$ is the model learned from training data. $P(C)$, the language model, is the prior probability for allowed sequence of characters. The language model used in the OCR system is a finite-lexicon word n-gram Markov model. The goal is to maximize likelihood term $P(X|C)P(C)$ since $P(X)$ is independent of C. More details about the OCR engine can be found in [20].

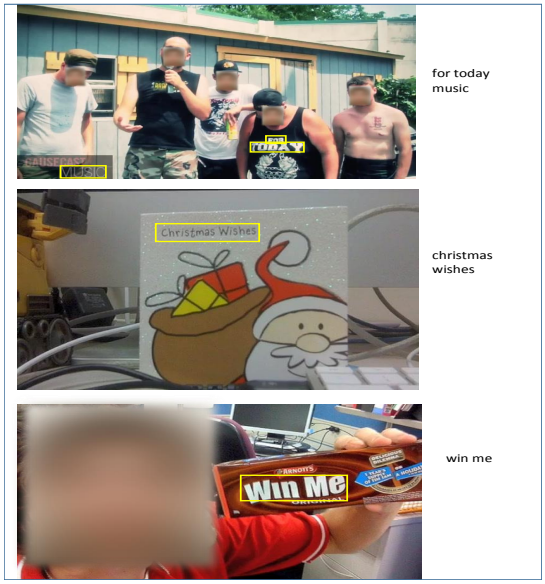


Fig. 4. Qualitative results of our algorithm

Table 1. OCR Word Recognition Performance

	Precision	Recall	F-score
[9]	0.045	0.234	0.076
Ours	0.147	0.370	**0.210**

5. EXPERIMENTS

We evaluated the performance of our system on a large consumer video dataset on two tasks.

Task 1) Text Detection and Recognition: We selected a subset of 1750 videos from the TRECVID MED dataset [21], created from consumer videos on web. Each video frame is annotated with text region bounding boxes and underlying words. These are unconstrained videos with varying backgrounds, text fonts and stroke widths which make them extremely challenging for text detection and recognition tasks.

Implementation details: HOG cell size and block size are set to 4 pixels and each candidate is resized to 32×32 before computing HOG features, resulting in 2025 feature dimensions. Gabor filters are computed for $\phi = \{0, \frac{\pi}{4}, \frac{\pi}{2}, \frac{3\pi}{4}\}$, $\lambda = \{5, 10, 12\}$ and $\sigma = \{1, 1.58, 1.87\}$ values, resulting in 36 dimensional Gabor features. 9 PLS dimensions are selected based on 5-fold-cross-validation on training data.

We sampled video frames uniformly at the rate of 2 fps and ran our text detection and recognition system. We compare the OCR output performance based on the proposed text detector with that based on the CRF based detector [9] in Table 1. We significantly improve both frame-level word precision and recall scores for OCR output compared to [9].

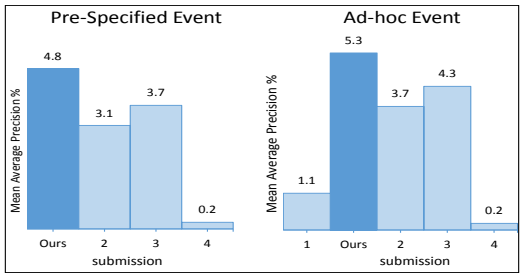
Additionally, to evaluate pixel level precision-recall for our text detection algorithm, we collected 596 images from these videos and divided them into 388 training and 208 testing images. We followed the same evaluation scheme described in [9] to compute precision-recall scores. Table 2 shows the performance comparison for text detection.

Task 2) Event Detection: We also include TRECVID

Table 2. Text Detection Performance

	Precision	Recall	F-score
[9]	0.7066	0.1444	0.2392
Ours	0.5209	0.3024	**0.3823**

Fig. 5. Performance of BBN's OCR-only system on TRECVID MED task (Ek100 condition) compared with other submissions.



Multimedia Event Detection (EK100 condition) [21] performance, based on only the video text content. The testing data consists of about 100,000 consumer-generated videos involving 20 pre-specified events and 10 additional ad-hoc events, along with a large set of background videos. The SVM classifier for each event is trained with 100 positive training videos and a set of background training videos. The BBN OCR-only system uses the video text detection and recognition output from the components described in this paper. The OCR decoding output word lattice for each video is converted into a vector where each dimension corresponds to one different word weighted by its expected count in the lattice and a revised inverse document frequency. Table 5 illustrates the OCR-only Multimedia Event Detection (MED) performances in the TRECVID 2013 MED evaluation, highlighting the competitiveness of our system.

6. CONCLUSION

We propose an end-to-end text detection and recognition system. The text detection component uses SVM classifier based on rich shape descriptors such as HOG, Gabor and edge features for improved performance, and leverages PLS technique for dimensionality reduction, leading to SVM speed improvement. We proposed a merging scheme which overcomes the mistakes of SVM classification step and preserves word boundaries. Extensive evaluation on a large dataset illustrates the efficacy of our approach in both pixel-level text detection and word recognition tasks.

Acknowledgement: This work is supported in part by the DARPA MADCAT program, and in part by the Intelligence Advanced Research Projects Activity (IARPA) via the Department of Interior National Business Center contract number D11PC20071. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright annotation thereon. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied , of the Department of Defense, IARPA, DoI/NBC, or the U.S. Government.

7. REFERENCES

[1] Ray Smith, "An overview of the tesseract ocr engine," in *Proc. Ninth Int. Conference on Document Analysis and Recognition (ICDAR)*, 2007.

[2] ABBYY (2013), "Abbyy mobile products. available: http://www.abbyy.com/mobile/accessed," 2013.

[3] Milyaev S, Barinova O, Novikova T, Lempitsky V, and Kohli P, "Image binarization for end-to-end text understanding in natural images," in *International Conference on Document Analysis and Recognition (ICDAR)*, 2013.

[4] Keechul Jung, Kwang In Kim, and Anil K. Jain, "Text information extraction in images and video: a survey," *Pattern Recognition*, vol. 37, no. 5, pp. 977–997, 2004.

[5] Serge Belongie Kai Wang, Boris Babenko, "End-to-end scene text recognition," in *International Conference on Computer Vision*, 2011.

[6] Jung-Jin Lee, Pyong-Hean Lee, Seong-Whan Lee, Alan Yuille, and Christof Koch, "Adaboost for text detection in natural scene," in *International Conference on Document Analysis and Recognition (ICDAR)*, 2011.

[7] Chucai Yi and Yingli Tian, "Text detection in natural scene images by stroke gabor words," in *International Conference on Document Analysis and Recognition (ICDAR)*, 2011.

[8] Boris Epshtein, Eyal Ofek, and Yonatan Wexler, "Detecting text in natural scenes with stroke width transform," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2011.

[9] Xujun Peng, Huaigu Cao, Rohit Prasad, and Premkumar Natarajan, "Text extraction from video using conditional random fields," in *International Conference on Document Analysis and Recognition (ICDAR)*, 2011.

[10] Maximally Stable Extremal, J. Matas, O. Chum, M. Urban, and T. Pajdla, "Robust wide baseline stereo from," in *In British Machine Vision Conference*, 2002, pp. 384–393.

[11] L. Neumann and J. Matas, "Real-time scene text localization and recognition," *2012 IEEE Conference on Computer Vision and Pattern Recognition*, vol. 0, pp. 3538–3545, 2012.

[12] Xuwang Yin, Xu-Cheng Yin, Hong-Wei Hao, and Khalid Iqbal, "Effective text localization in natural scene images with mser, geometry-based grouping and adaboost," in *International Conference on Pattern Recognition*, 2012.

[13] Cunzhao Shi, Chunheng Wang, Baihua Xiao, Yang Zhang, and Song Gao, "Scene text detection using graph model built upon maximally stable extremal regions," *Pattern Recognition Letters*, vol. 34, no. 2, pp. 107 – 116, 2013.

[14] Navneet Dalal and Bill Triggs, "Histograms of oriented gradients for human detection," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2005.

[15] C. Harris and M. Stephenes, "A combined corner and edge detector," in *Alvey Vision Conference*, 1988.

[16] J. Canny, "A computational approach to edge detection," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 8, pp. 679–714, 1986.

[17] H. Wold, "Partial least squares," *In S. Kotz and N. Johnson, editors, Encyclopedia of Statistical Sciences*, vol. 6, pp. 581–591, 1985.

[18] Chih-Chung Chang and Chih-Jen Lin, "LIBSVM: A library for support vector machines," *ACM Transactions on Intelligent Systems and Technology*, vol. 2, pp. 27:1–27:27, 2011, Software available at http://www.csie.ntu.edu.tw/ cjlin/libsvm.

[19] Nobuyuki Otsu, "A threshold selection method from gray-level histograms," *IEEE Transactions on Systems, Man and Cybernetics*, pp. 62–66, 1979.

[20] Premkumar Natarajan, Zhidong Lu, Richard Schwatrz, Issam Bazzi, and John Makhoul, "Multilingual machine printed ocr," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 15, 2001.

[21] Paul Over, George Awad, Martial Michel, Jonathan Fiscus, Greg Sanders, Wessel Kraaij, Alan F. Smeaton, and Georges Quenot, "Trecvid 2013 – an overview of the goals, tasks, data, evaluation mechanisms and metrics," in *Proceedings of TRECVID 2013*. NIST, USA, 2013.