2011 International Conference on Document Analysis and Recognition

Text Extraction from Video Using Conditional Random Fields

Xujun Peng, Huaigu Cao, Rohit Prasad and Premkumar Natarajan
Raytheon BBN Technologies, Cambridge, MA 02138, USA
{*xpeng, hcao rprasad, pnataraj*}*@bbn.com*

***Abstract*—In this paper, we describe an approach to extract text from broadcast videos. Candidate blocks are detected based on edge extraction results. Corners and geometrical features are used for the purpose of initial classification which is carried out by using a support vector machine (SVM). Considering the spatial inter-dependencies of different regions in the image, we propose a novel conditional random field (CRF) based framework which integrates the outputs of SVM into the system to improve the accuracy of labeling for blocks. The experimental results show that the proposed system achieves reliable performance for text detection/extraction from videos.**

***Keywords*-Text; Conditional Random Field; Video**

I. Introduction

Nowadays, images and videos on the internet and in databases are growing at an astounding pace. However, despite the vast information these multimedia resources provided for the public, it is still a challenging task to organize and retrieve this information automatically [9]. Texts in images or videos, which contain useful high level information, offer us an excellent opportunity for annotation, indexing, understanding and information retrieval. Thus, text detection, which locates text regions in images and videos, plays an important role in the design of system for image/video understanding and retrieval.

Although lots of efforts have been taken to detect and extract text from images and videos, there are still many constraints, such as low contrast, complex background, bad illumination, different text layout and font size, etc, that limit the performance of video text detection.

Generally, text detection from videos can be roughly categorized into three groups: connected component (CC) based methods, texture-based methods and edge-based methods. Connected component based methods divide an image or video frame into small regions by merging neighboring pixels according to their colors or grayscale intensities and then classify the candidate regions as text or background [15]. Because in most circumstances texts have varied colors or grayscales within the video, this type of methods can not provide a satisfying performance for text detection.

By considering the text area as a special kind of texture, texture based methods use Gabor filters, wavelet decomposition, fast Fourier transform (FFT) and discrete cosine transform (DCT) to analyze different regions in the video frame and label them as text or background. In [12], Crandall et al. segmented the entire image into 8×8 blocks and extracted the DCT coefficients for each of them. The final text locations were obtained by using a single threshold based method and an iterative greedy grouping algorithm. Similarly, Kim divided the video frame into small windows and used gray values of the raw pixels in the window as the input of SVM texture classifier for text detection [11]. The potential problem of texture based techniques is the high computational complexity of the FFT, DCT, and other transforms.

In order to overcome the drawback of texture analysis, many light-weight features were used for text detection. Phan et al. [14] proposed a method which filtered images using a Laplacian mask and detected the text regions based on the maximum gradient value on the filtered image. Based on the observation that corners are frequent patterns in text regions, Zhao et al. [7] proposed a corner based technique which extracted and dilated Harris points in images and discriminated text regions from background using a decision tree classifier. In [8], a transient color based method was proposed by W. Kim and C. Kim where a transition map was generated on the video scenes for text detection.

Based on the assumption of rich edge information within the text areas, edge based text detection methods extract lines of characers/texts and locate text regions according to these lines. In [10], Epshtein et al. suggested a stroke width transform algorithm which was a variation of edge based method to extract text from natural scenes.

To combine the advantages and avoid the shortcoming of previous methods, Pan et al. [2], [3] used multiple spatial and geometrical features to detect text in videos. In addition, they applied a conditional random field-based method to label text areas.

In this paper, we present a conditional random field (CRF) based approach to detect the text lines from video frames. The proposed system contains three basic stages as shown in Fig. 1: Text block extraction based on edges, support vector machine (SVM) prediction and CRF labeling for text regions and the text line aggregation. We organize the rest of paper as follows. Section II introduces the preprocessing

[1] The views expressed are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

[2] Distribution Statement “A” (Approved for Public Release, Distribution Unlimited).

1520-5363/11 $26.00 © 2011 IEEE
DOI 10.1109/ICDAR.2011.208

step which includes text block and feature extraction for video scenes. Section III describes the method which integrates the prediction of SVM based classification into the CRF-based labeling framework and shows the approach to aggregate isolated text regions into a text line. The detailed experimental set up and results are covered in section IV and the conclusions are covered in section V.

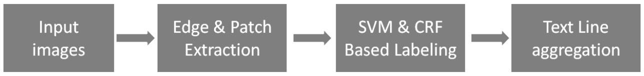


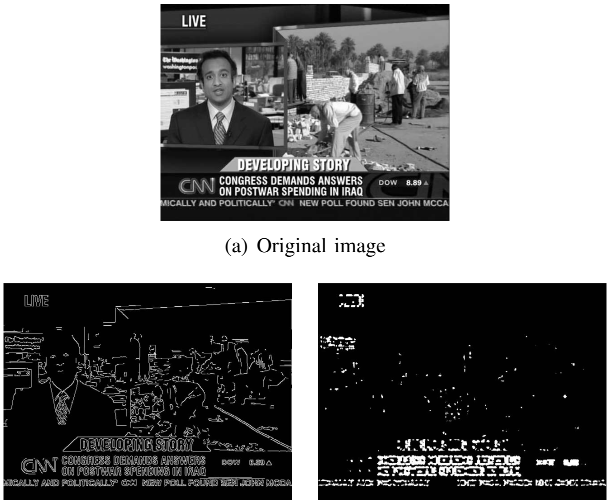
Figure 1. The flowchart of the text detection system.

II. Preprocessing

Our pre-processing consists of two steps: (i) text block extraction and (ii) feature extraction.

A. Text block extraction

Edges are the features that are salient and robust for pattern classification, especially for text and characters, which often have high contrast with the background. Prior to feature extraction and text detection, the video frames are converted into gray-scale images and enhanced by using method described in [19] to ensure the text strokes have strong edges. The edges of the image from videos are detected using the Canny operator [16]. From Fig. 2(b), we can see that all edges from texts are detected but with plenty of noise (false alarms) because of complex backgrounds.


LIVE
DEVELOPING STORY
CNN CONGRESS DEMANDS ANSWERS ON POSTWAR SPENDING IN IRAQ
DOW 8.89
MICALLY AND POLITICALLY CNN NEW POLL FOUND SEN JOHN MCCA

(a) Original image

(b) Detected edges

(c) Detected corners

Figure 2. An example of edge detection using Canny operator and corner detection using Harris detector.

For each connected component on the edge map image, its bounding box is calculated and the original content within the bounding box of each connected component is considered as a candidate text block or patch. Blocks whose size is bigger than a predefined threshold t_h or smaller than a threshold t_l are eliminated as noise or line.

B. Feature extraction

A set of features are considered for classification of a given candidate block into one of two classes: text and background.

In order to extract corner related features, the Harris corner detection [17] is carried out on the original video image and the corner map image is binarized using a predefined threshold. Fig. 2(c) shows the result example of Harris corner extraction where white dots indicate the corner points.

Based on the analysis that strokes of texts have high frequency of endings, intersections and bend lines which cause more corners in the text region than background area, we compute the ratio of corners to edges for each block. Assuming that the size of a given candidate block b_i is $w \times h$, the ratio is calculated according to Equation. 1:

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

where x and y run over the area of the candidate block b_i, $C(x,y)$ denotes the intensity of the corner map and $E(x,y)$ denotes the intensity of the edge map of the input image which is obtained using Canny operator as described in previous section.

The standard deviation is computed from the output of Gabor filtering for each candidate block to extract texture features. Gabor filters are band-pass filters which are modulations of a complex sinusoidal and Gaussian function and can be used to detect the direction of strokes. The 2-D Gabor filters are formulated as:

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

where $R_1 = x\cos\phi + y\sin\phi$ and $R_2 = y\cos\phi - x\sin\phi$, λ is the wavelength of a Gabor filter, ϕ denotes the orientation of a Gabor filter and σ_x and σ_y control the standard deviation of Gabor filter. In this paper, two orientations ϕ are used in our experiments which detect horizontal and vertical strokes.

In addition, considering each block/patch as a single unit in our system, we extract its spatial location x_i and y_i with respect to the image as features. The width w and height h of each candidate block are also calculated. Each feature is normalized in the range of -1 to 1 using a linear mapping.

III. Block Labeling and Merging

A. SVM Classification

To label the candidate block as text or background, we initially predict the probability of each block to be text using an SVM.

Support vector machine (SVM) is widely used for pattern classification which is specifically designed for two-class problems. Given a set of training samples along with their labels $\Phi = \{(x_i, y_i)|i = 1, 2, \cdots, m\}$, where x_i belongs to R^n of the feature space, y_i belongs to the label set $(+1, -1)$ and the m indicates the size of training set, the linear SVM finds an optimal separating hypothesis $H : x \to y$, which maps the feature x of a given block to a label y.

Normally, a linear SVM classifier is defined as:

$$\begin{aligned} H(x) &= sign\,\{f(x)\} \\ &= sign\left\{\sum_{i=1}^{n} a_i y_i (x_i \times x) + b\right\} \end{aligned} \tag{3}$$

where $sign\{\}$ is the sign function, a_i is the Lagrange multiplier, x_i is a supporting vector from training set and x is a testing sample.

A non-linear SVM classifier is simply carried out by mapping the data into a high dimensional space where linear separation can be performed.

While an SVM is often used to output the label of testing samples, it can also be used to estimate a posterior probability of a given likelihood by using a sigmoid function:

$$Pr(y = 1 \mid x) \approx P(f(x)) \equiv \frac{1}{1 + exp(Af(x) + B)} \tag{4}$$

where the optimal parameters A and B are determined by solving a regularized maximum likelihood problem [1]. In this paper, we denote $Pr(y_i = 1 \mid x_i)$ as p_i to measure the the confidence of a given block to be text and $q_i = 1 - p_i$ to predict its probability to be background.

B. Conditional Random Fields

Conditional random fields (CRFs) are discriminative graph models which are designed for labeling tasks such as text identification [5] and document image segmentation [4]. The motivation to use CRFs to label the text region from video frames arises from the spatial inter-dependencies of different areas in images. For example, text blocks are sequential from left to right. By considering neighboring information of blocks, isolated noises among text blocks can be easily removed which leads to more satisfactory labeling results.

Unlike other generative graphical models such as Markov random fields (MRFs) which require specifying the likelihood function, CRFs have a more flexible formulation [6]. More formally, let $X = \{x_i\}$ be the observed features from candidate blocks, and $Y = \{y_i\}$ be random variables over corresponding labels. The joint distribution over the label y_i given an observation x_i has the form:

$$p(y_i \mid x_i) \propto \exp\left(\lambda A(y_i, X) + \mu \sum_{(i,j)\in E} I(y_i, y_j, X)\right) \tag{5}$$

where function $A(\cdot)$ is called associated potential which measures the confidence of label y_i with observations, function $I(\cdot)$ is interaction potential which tends to smooth labels over entire graph G, λ and μ are parameters that control the influence from observations and neighboring nodes to center node i, and $(i, j) \in E$ means neighboring nodes of node i that are connected by edges E in the graph G.

In our work, we use the topology shown in Fig. 3(a) for our CRFs. By making a Markov assumption, each gray node y_i in the hidden layer exclusively corresponds to a detected block in the image and connects to its four nearest neighbors and their corresponding observations. In real images, the neighbor system of blocks is determined by Euclidean distance between them but may not necessarily be located as a grid as shown in Fig. 3(a).

To integrate the predicted confidence of blocks into CRFs framework, we define the associated potential as:

$$A\,(y_i, X) = \sum_{j\in N} p_j \exp\left(-|d_{i,j} \cdot \cos(\theta_{i,j})|\right) \tag{6}$$

where j runs over neighbors of node i including itself, p_j is the posterior estimated by the SVM for node j, $d_{i,j}$ is the spatial Euclidean distance between node i and j, and θ is the angle between centers of node i and j.

The idea behind equation 6 is that if two neighboring nodes are close to each other and their separation is mostly horizontal, they have more influences on each other.

The interaction potential in the proposed system is defined in a way to encourage neighboring nodes with the same labels:

$$I\,(y_i, y_j, X) = \exp\left(y_i \cdot y_j \cdot |x_i - x_j|\right) \tag{7}$$

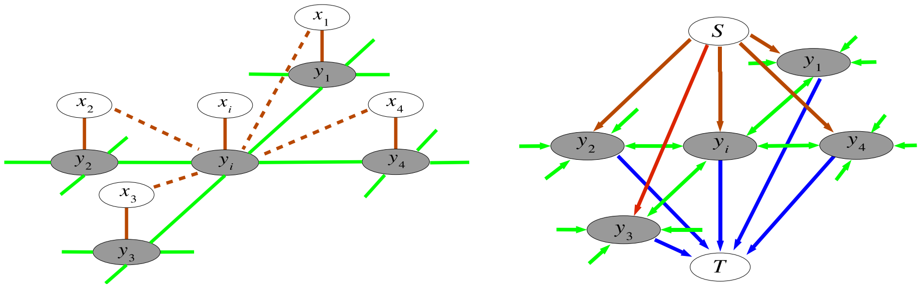
where $|x_i - x_j|$ measures the distance of observations of two neighboring nodes i and j in feature space, y_i and y_j are labels assigned to two nodes. From equation 7 we can see that if two neighboring nodes have the same label (-1 or $+1$), the potential function has a value bigger than 1. Otherwise, its value is smaller than 1.

The overall potential function of the graph G is maximized by a min cut/max flow algorithm where the flow from the source to each node is $A(y_i, X)$, the flow from each node to the sink is $1 - A(y_i, X)$ and the flow from neighboring node j to node i is $I(y_i, y_j, X)$ as shown in Fig. 3(b). More details of min cut/max flow algorithm can be found in [13].

C. Text Line Aggregation

The text blocks that are labeled using CRFs are isolated connected components in our system. To further extract text line from videos, a heuristic reasoning scheme is taken to merge isolated text blocks into lines based on the assumption that texts appear in a linear manner.

Assuming the spatial location and size of the bounding box for a given text block b_i are denoted as $[x_i, y_i, w_i, h_i]$,



(a) The topology of the CRF

(b) The graph of max-flow

Figure 3. (a). Each grey node y_i which is a label for a text block connects to its four nearest neighbors and their corresponding observations which are associated with their features. Edges between any pair of nodes indicate their similarity or dependency. (b). Node S is the source where flows go out only and T is the sink where flows go into only. Each gray node corresponds to the hidden node of CRFs. Directed edges indicate the direction of flows.

the bounding box of each text block is extended h_i in right direction if its closest right neighboring text block b_j satisfies: $|h_i - h_j| < 0.3 \cdot \min(h_i, h_j)$ and $|h_i + y_i - (h_j + y_j)| < \tau$, where τ is a pre-defined threshold which is set to be 5 in our experiments. The final text line is extracted by calculating the bounding box of all overlapped bounding boxes.

IV. Experiments

For experimental purpose, we collected a total number of 660 images which were taken from broadcast videos where the text had different colors, sizes and styles on the varied backgrounds. The resolution of these images was 640×480 and the text language was English. The entire data set was separated into training set with 331 images and testing set with the remaining 329 images.

All original images were converted from RGB to gray-scale and candidate blocks and their corresponding features were extracted using the method described in section II. In the training phase, a block from the training set was determined as a text block if 90% of its area fall into the text region. Otherwise, this block was considered as a background block. A public tool LIBSVM [18] was used to train the SVM for our system, using a Radial Basis kernel.

In the testing phase, the label and posterior of each candidate block were predicted by the SVM initially. This information was integrated into the CRFs framework as described in section III-B and the optimal label of each block was obtained using min cut/max flow algorithm [13]. The final region of text line was detected by merging isolated text blocks as introduced in section III-C.

We measured the performance of the proposed system by precision and recall metrics on two levels: block level and pixel level.

At the block level, we counted the number of correctly detected text blocks (CB), false detected text blocks (background blocks labeled as text blocks by the system, FB), and missed detected blocks (text blocks labeled as background blocks by the system, MB). Thus, the precision and the recall were defined as:

$$P = \frac{\#CB}{\#CB + \#FB} \tag{8}$$

$$R = \frac{\#CB}{\#CB + \#MB} \tag{9}$$

At the pixel level, we defined the precision and recall using following equations:

$$P = \frac{R_e \cap R_t}{R_e} \tag{10}$$

$$R = \frac{R_e \cap R_t}{R_t} \tag{11}$$

where R_e is the regions of detected text lines and R_t is the ground truth areas of text line, $R_e \cap R_t$ means the overlapped areas of detected text line and ground truth.

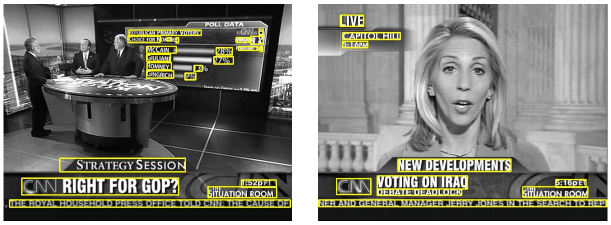
Table I shows the performance of an SVM only method and the proposed method for text detection from which we can see that the recall and precision at both block level and pixel level increased after using CRF based text detection method.

Fig. 4 shows result examples from the data set where text were detected even on a complex background.

The errors in our system mainly occurred because characters which touched with other long lines were mis-labeled as background because of their geometrical properties. The characters with large size appeared as background in images were typically classified as background also because we had limited number of which in the training set. The problem of large font size can be solved by adding more such samples into the training set.

V. Conclusions

In this paper, we present a conditional random field-based method to locate the areas of text from video frames. To integrate an SVM into the framework of CRFs, we use the estimated posterior from an SVM as a feature in the potential function of a CRF, where a distance-based function is used to compute optimal labels for text blocks. Experimental



(a) (b)

Figure 4. Examples of detected text line of broadcast videos from the system.

	SVM		CRF	
	Precision	Recall	Precision	Recall
Block Level	83.5%	95.6%	85.1%	96.6%
Pixel Level	93.4%	87.1%	94.3%	87.4%

Table I
RESULTS OF TEXT DETECTION.

results show that the proposed method out-performs the single SVM-based method. Our future work includes the exploration of new text block and feature extraction method and use of new classifier other than SVM.

ACKNOWLEDGEMENTS

This paper is based upon work supported by the DARPA MADCAT Program.

REFERENCES

[1] Hsuan-Tien Lin, Chih-Jen Lin and Ruby Weng, A note on Platts probabilistic outputs for support vector machines, Machine Learning, 68(3), pages 267-276, 2007.

[2] Y. Pan, X. Hou and C. Liu, A Hybrid Approach to Detect and Localize Texts in Natural Scene Images, IEEE Transactions on Image Processing, 20(3), pages 800-813, 2011.

[3] Y. Pan, X. Hou and C. Liu, Text Localization in Natural Scene Images Based on Conditional Random Field, 10th International Conference on Document Analysis and Recognition, pages 6-10, 2009.

[4] S. Nicolas, J. Dardenne, T. Paquet and L. Heutte, Document Image Segmentation Using a 2D Conditional Random Field Model, 9th International Conference on Document Analysis and Recognition, 1, pages 407-411, 2007.

[5] S. Shetty, H. Srinivasan, S. N. Srihari and M. Beal, Segmentation and labeling of documents using Conditional Random Fields, Proc. Document Recognition and Retrieval IV, Proceedings of SPIE, 6500U-1-11, 2007.

[6] J. Lafferty, A. Macullum and F. Pereira, Conditional random fields: Probabilistic models for segmenting and labeling sequential data, 8th International Conference on Machine Learning (ICML), pages 282-289, 2001.

[7] X. Zhao, K. Lin, Y. Fu, Y. Hu, Y. Liu and T. S. Huang, Text From Corners: A Novel Approach to Detect Text and Caption in Videos, IEEE Transactions on Image Processing, 20(3), pages 790-799, 2011.

[8] Wonjun Kim and Changick Kim, A New Approach for Overlay Text Detection and Extraction From Complex Video Scene, IEEE Transactions on Image Processing, 18(2), pages 401-411, 2009.

[9] P. Shivakumara, W. Huang, T. Q. Phan and C. L. Tan, Accurate video text detection through classification of low and high contrast images, Pattern recongition, 43(6), pages 2165-2185, 2010.

[10] B. Epshtein, E. Ofek, and Y. Wexler, Detecting text in natural scenes with stroke width transform, IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 2963-2970, 2010.

[11] Kwang In Kim, Keechul Jung and Jin Hyung Kim, Texture-based approach for text detection in images using support vector machines and continuously adaptive mean shift algorithm, IEEE Transactions on Pattern Analysis and Machine Intelligence, 25(12), pages 1631-1639, 2003.

[12] David Crandall, Sameer Antani, Rangachar Kasturi, Extraction Of Special Effects Caption Text Events From Digital Video, International journal on document analysis and recognition, 5(2-3), pages 138-157, 2003.

[13] Y. Boykov and V. Kolmogorov, An Experimental Comparison of Min-Cut/Max-Flow Algorithms for Energy Minimization in Vision, IEEE Trans. Pattern Recognition and Machine Intelligence, 26(9), pages 1124-1137, 2004.

[14] Trung Quy Phan, P. Shivakumara and Chew Lim Tan”, A Laplacian Method for Video Text Detection, 10th International Conference on Document Analysis and Recognition(ICDAR), pages 66-70, 2009.

[15] A.K. Jain, and Y. Bin ”, Automatic text location in images and video frames, 14th International Conference on Pattern Recognition, 2, pages 1497-1499, 1998.

[16] J. Canny, A Computational Approach To Edge Detection, IEEE Trans. Pattern Analysis and Machine Intelligence, 8, pages 679-714, 1986.

[17] C. Harris and M. Stephenes, A combined corner and edge detector, Vlvey Vision Conference, pages 147-151, 1998.

[18] Chih-Chung Chang and Chih-Jen Lin, LIBSVM: a library for support vector machines, http://www.csie.ntu.edu.tw/cjlin/libsvm, 2001

[19] X. Peng, S. Setlur, V. Govindaraju and R. Sitaram, Binarization of Camera-Captured Document using A MAP Approach, Proc. SPIE of Document Recognition and Retrieval XVIII (DRR), 7874, pages 78740R-1-8, 2011.