2012 International Conference on Frontiers in Handwriting Recognition

A New Method for Text Verification based on Random Forests

Yang Zhang, Chunheng Wang, Baihua Xiao, Cunzhao Shi
State Key Laboratory of Intelligent Control and Management of Complex Systems
Institute of Automation, Chinese Academy of Sciences
Beijing, China
{yang.zhang, chunheng.wang, baihua.xiao, cunzhao.shi }@ia.ac.cn

***Abstract*—Text in image or video frames contains a lot of high-level semantics which can be useful for multimedia indexing, management. Coarse text detection results may contain many false alarms, which makes it necessary to eliminate the false alarms for further recognition. As text has distinct textural features, texture-based classifier such as SVM, MLP and Adaboost has been used to classify the detection regions as text or non-text region. In this paper, a random forests based method for text verification is proposed. The reason of choosing random forests lies in: 1) its ability of maintaining accuracy in small labeled dataset and 2) its good performance in unbalanced dataset as in the case of unbalanced text and non-text distribution. Furthermore, we propose to merge different random forests trained with different kinds of features to improve the accuracy of classification. The comprehensive experimental results show that our methods are effective.**

Keywords-component; text verification; random forests; RF; GSC, MsLBP,EOH

I. INTRODUCTION

With the development of internet and digital products, multimedia information has a tremendous increase, which leads to an urgent demand for automatic semantic information indexing and retrieval systems. Textual information in images and videos proves to be an important source of high-level semantics, which can help the computer to understand the content of images and videos. Text detection and extraction in images and videos are very important for the fields of automatic annotations, indexing and parsing of images and videos. Many text detection and location methods have been proposed in recent years [1,2]. Most [5,7,16]of text detection or location algorithms can be concluded a two stage scheme shown in Fig 1.

For any detection algorithm, it is hard to ensure there are no flase alarms, so efficient text verification is important and necessary. For text verification, it also can be concluded to learning-based method [4-9] and heuristic-based method [3]. The method based heuristic has the advantages of easy and intuitive, but it is hard to remove non-text line with complex background. Learning-based method is more robust for distinguishing the text line and non-text line with complex background. Many features have been proposed to describe the text line image and train the classifier for text verification and refinement. Some of them are point-based, that is, the features are extracted based on the special pixel point or some kind of transformation of these points. But different texts under different background show quite different in style, color and structure, these point-based [5,6] features can hardly represent the features of text lines in common. Region-based [4,8,9] features are now more popular, which are more robust to describe the structure of text region and have the lower computation cost. And SVM [5,6,9] classifier is the most commonly used method for text verification as its strong generalization capability for small dataset classification. Random forests (RF) [10] have been applied to object recognition and tracking. The advantage of RF is that they are much faster in training and testing than traditional classifier (such as an SVM) and different cues can be effortlessly combined for classification [11,12]. As RF maintains accuracy when a large proportion of the data are missing and balances error in data sets with unbalanced class population, it is suitable for text verification.

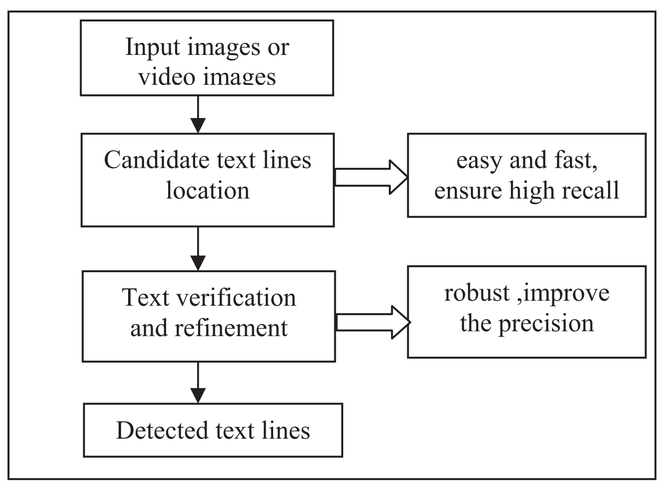


Fig. 1 two-stage scheme of text detection algorithms

In this paper, we would exploit the performance of random forests for text verification. And to combine different features with random forests trained with different kinds of features, we can improve the accuracy of

978-0-7695-4774-9/12 $26.00 © 2012 IEEE
DOI 10.1109/ICFHR.2012.163

classification. Experimental results demonstrate that random forests are suitable for text verification, superior or comparable with SVM and it can improve the accuracy of classification by merging different kinds of feature.

The rest of this paper is organized as follows: section 2 gives an overview of features for text verification. The random forests classifier is introduced in Section 3. We report the construction of datasets and related experiment results and analysis in Section 4. This paper concludes with a discussion and conclusions in Section 5.

II. RONDOM FORESTS CLASSIFIER

Random forests (RF) have become a popular technique for classification, prediction, study variable importance, variable selection, and outlier detection. There are numerous application examples of RF in a variety of fields. Compared to transitional classifiers such as SVM, they are much faster in training and testing. And they also enable different features to be combined efficiently. The extensive discussion of RF can be find in[10]. In this paper, we will only consider it as a binary classifier for text verification.

A. *Feature extraction*

Researchers make use of features to decide whether a pixel or block of pixels belongs to text or not. From that the features can be classified into two kinds: point-based and region-based. Many point-based features are proposed such as gray scale feature for text detection by Kim[6] and normalizes gray scale feature for text verification by Jung[13], distance map and constant gradient variance future by Chen[5] and Li et al.[14] used a neural network to extract text block in Haar wavelet feature space. Chen[8] and Shehzad[15] used the Adaboost to classify the text and non-text block with large features set such as mean difference, standard deviation and histogram of oriented gradients. Pan et al.[4] proposed feature pool for text region classification, which is composed of histogram of orientation Gradient and multi-scale Local Binary Pattern (MSLBP). About feature selection for text verification, Wang[9] proposed that the selected features should minimize the influence of background as far as possible and be expressive enough for the texts varied in structure. They proposed a new block partitioned feature for text verification which includes gray scale contrast feature (GSC) for minimizing the influence of background and edge orientation histogram (EOH) feature for describing the text structure.

Although point-based features do perform well for text detection and refinement, but that depends on database and text background. In our experiments, we used region-based feature for text verification, and GSC, EOH and MSLBP are used for comparison experiments which perferm well both on scene and video image datasets.

B. *Growing the Trees*

The trees here are binary and are constructed in a top-down manner. The algorithm is described as follows:

- Let the number of training cased be *N*, and the number of variables in the classifier be *M*.
- m input variables are used for decision making at each node of the tree; m should be much less than *M*.
- Choose a training set for this tree by choosing *n* times with replacement from all *N* available training cases (i.e. take a bootstrap sample). Use the rest of the cases to estimate the error of the tree, by predicting their classes.
- For each node of the tree, randomly choose *m* variables on which to base the decision at that node. Calculate the best split based on these *m* variables in the training set.
- Each tree is fully grown and not pruned (as may be done in constructing a normal tree classifier).

C. *Random Forests Classification*

For classification a new sample is passed down the tree until it reaches a leaf node which is assigned the label of train sample. This procedure is iterated over all trees in the ensemble, and the average vote of all trees is reported as random forest classification.

D. *Merging Different Kinds of Features*

To combine the features, we wish to enable the classifier to be selective for different kinds of features. Since some text and non-text can be classified by the gray scale contrast, and other may be classified correctly just by texture features, Added different features is no optimal. An alternative way to merge the different kinds of feature is to build a forest for each kind of feature, and merge them for the classification. The number of trees in each RF for different kinds of feature is fixed by cross validation on the training dataset.

E. *Text Verification Method*

In this paper, the weighted method for verification scheme [5] is used to verify the text lines. Here we just think about the horizontal text lines. First the text lines are resized to a uniform height. Then we slide a size-fixed window with a step from left to right in each normalized text line and compute the confidence of each window. The confidence Conf(*R*) of a text line *R* was defined as:

$$Conf(R) = \mathrm{sgn}\left(\sum_{i=1}^{l} f(x_i^R) * \frac{1}{\sqrt{2\pi}\sigma_0} * e^{-d_i^2/2\sigma_0^2}\right) \quad (1)$$

Where d_i is the distance from the geometric center of the *i*th sliding window to geometric center of candidate text R. σ_0 is a scale factor depending on the text line length, and $f(x_i^R)$ is the classification probability of *i*th sliding window region got by classifier (SVM or RF). How to set the values of all parameters in our experiments are showed in Section 4.

III. EXPERIMENTS AND ANALASIYS

To exploit the performance of RF for text verification, we do groups of comparisons experiments with SVM and with different features. The following part will introduce the experiments details. Our experiments are conducted at a personal computer with Core i3 CPU 1.99GHZ processer and 2G main memory. The program language is matlab mixed with C++.

A. Evaluation Measurements

In our experimental results, we adopt the following three measurements to evaluate the performance of classifier: correctly classification rate (CCR) which means for positive text line classifier can correctly verify it as text, false classification rate (FCR) which means for non-text line classifier also verify it as text and the precision rate of classification (PRC) which means the total precision rate on all the samples. The classifier should keep high CCR, high PRC and low FCR.

$$CCR = \frac{N_{ct}}{N_{text}} \times 100\% \tag{2}$$

$$FCR = \frac{N_{fnt}}{N_{non-text}} \times 100\% \tag{3}$$

$$PRC = \frac{N_{ccsamples}}{N_{all-samples}} \times 100\% \tag{4}$$

Where N_{ct} and N_{text} denote the number of correctly classified texts and the total number of text in the database. N_{fnt} and $N_{non-text}$ denote the number of wrongly classified non-text and the total number of non-text. $N_{ccsamples}$ and $N_{all-samples}$ denote the number of all correctly classified samples and the number of all the samples.

B. Traing and Testing Samples Details

In our experiments, images of text from three languages are used: English, Chinese, and Japanese. The English dataset are collected from the ICDAR text detection competition, which mainly include text embedded in the natural scene and advertisement image. The Chinese and Japanese dataset are grabbed frames from Videos, including News, TVs. In our all experiments, the text lines are horizontal and the width is at least twice longer than the height. The height of text lines are varied from 9 to 70 pixels. From the English database we select 704 positive text lines and 484 non-text lines as the training data, 200 positive text lines and 200 non-text lines as testing data. From the Chinese database we select 400 positive text lines and 500 non-text lines as the training data, 100 positive text lines and 100 non-text lines as testing data. From the Japanese database we select 200 positive text lines and 420 non-text lines as the training data,100 positive text lines and 400 non-text lines as testing data. Some of the text lines and non-text lines are shown in Fig.2.

(c) text lines of Japanese

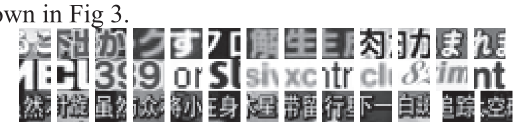
(d) non-text lines

Fig.2 Samples of text and non-text lines

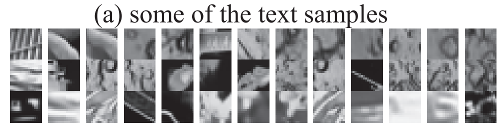
C. Experimental Parameters Set

In previous text verification methods, designed positive of training data have two ways. One is the single text with no noise, another is the text line in the images are labeled instead of single text, and then a sliding window is used to extract the texts as positive training data. These two methods have respective advantages: the first one trains the classifier more precise and the latter is more suitable for the verification mechanism. In our experiments, we adopt the latter and try to ensure that the texts are distributed proximate in positive text lines, which means that sliding window can include more text and less background. Some of the text and non-text are shown in Fig 2.

In our experiments, horizontal text lines and background images are collected for verification. The height of text lines is normalized to 16. And we set the height of sliding window size as 16, the width as 16.The sliding step is set as half of the width of window. Some of train samples are shown in Fig 3.



(a) some of the text samples



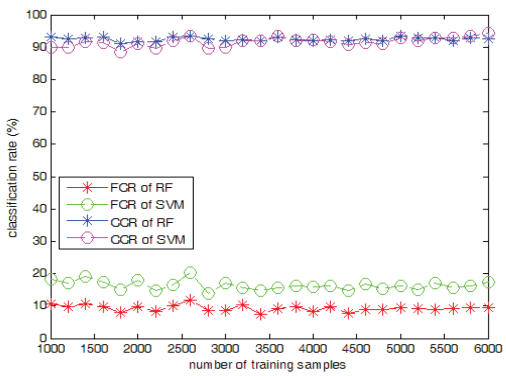
(b) some of the non-text samples

Fig.3 samples for training classifier

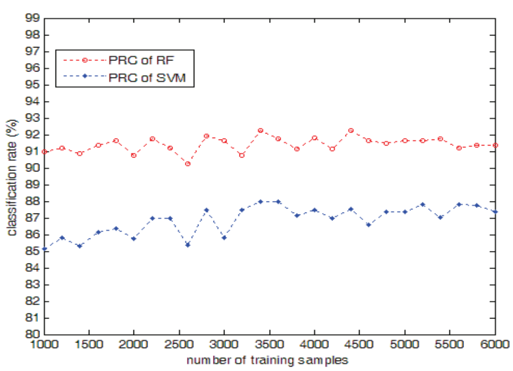
D. Classification Performance: RF vs SVM

To exploit the performance of random forests on text verification, we compare it with the SVM classifier [9], which is the most commonly used for text verification. Here we use the same features: GSC, EOH and MSLBP.

With the sliding window scheme, 10705 positive samples and 8774 negative samples are collected. We train the classifiers with incremental train samples from 1000 to 6000, positive and negative samples each takes half, which are randomly selected from the total training dataset. In this experiment, we adopt GSC+EOH [9] as the features for training and classification. And then test dataset composed of 400 text lines and 700 non-text lines is classified with these classifiers. The number of trees in RF is set 120. We do the experiments five times and the results are their average. The results will be shown in Fig 4.



(a) FCR and CCR



(b) Precision rate of classification

Fig.4 Performance of RF and SVM on incremental training samples

From above results, we can see that RF has the obvious advantage on the FCR and is comparable with SVM on CCR even superior on less training dataset. As we know that the non-text lines samples are infinite, although non-text line samples are collected on different resources, which cannot be enough for summarizing the non-text line data. RF maintains accuracy when a large proportion of the data is missing and balances error in class population unbalanced data sets. And the training and testing time of RF mainly depend on the tree number while SVM depends on the number of training samples. In our experiments, the training and testing time of SVM are twice as the RF when the training samples are 10000. The experimental results show that RF is suitable for text verification.

E. Performance of Merging the features by RF

All kinds of features have been proposed for text verification. They do show the strong discrimination ability on some dataset, but how to combine all kinds of feature is less studied. Most of the combination methods are to add the different features as one, which is unreasonable. By voting mechanism RF can combine different kinds of feature easily and it is more efficient than adding the features.

In this experiment, we choose the GSC, EOH and MSLBP as the features, which have been proved effective for text verification. Here all the classifiers are random forests, but with different kinds of feature and different kinds of combination. GSC+EOH means these two kinds of feature are adding as one feature, meanwhile GSC(r+)EOH means these two kinds of features are combined with random forests as described in Section 3. In this experiment, the number of trees in RF is 120. The experimental results are shown in Tab 1.

features \ performance	CCR	FCR	PRC
GSC	65.34%	12.88%	74.53%
EOH	71.45%	22.54%	75.02%
MSLBP	76.18%	21.22%	77.71%
GSC+EOH	90.34%	10.59%	90.64%
GSC (r+)EOH	91.67%	9.12%	92.49%
GSC+MSLBP	82.61%	13.18%	85.12%
GSC(r+)MSLBP	89.87%	10.53%	89.96%
EOH+MSLBP	76.39%	19.03%	79.11%
EOH(r+)MSLBP	83.54%	18.75%	82.16%
GSC+EOH+MSLBP	81.33%	13.76%	84.24%
GSC(r+)EOH(r+)MSLBP	91.24%	8.48%	91.54%

Tab 1. Performance of different features with RF

From the results, we can see that GSC feature performs well for FCR which means it can remove the false alarm (background image) effectively, and EOH and MSLBP feature perform well for CCR, which means they can keep high recall for text line images. In a word, it doesn't perform well when single feature are used. Adding features can improve the performance but more efficient by combining the features with RF.

IV. CONCLUSION AND FUTURE WORK

In this paper, we exploit the performance of RF for text verification. From the experimental results, RF can achieve comparable or better performance than SVM. And different kinds of feature can be merged easily to improve the accuracy of classification by RF.

Text verification is one step of text detection and extraction; it can effectively improve the detection precision and reduce the false alarm. But to get the text information from images, we also need to segment the texts from the complex background and recognize them. That will be a challenging work.

ACKNOWLEDGEMENTS

This work is supported by the National Natural Science Foundation of China under Grant No. 60835001, No. 60933010 and No.60802055.

REFERENCES

[1] K. Jung, K. In Kim, and A. K. Jain, "Text information extraction in images and video: a survey," *Pattern Recognition,* vol. 37, pp. 977-997, 2004.

[2] Jing Zhang, Rangachar Kasturi. Extraction of Text Objects in Video Documents: Recent Progress. In Proceedings of Document Analysis Systems'2008. pp.5~17.

[3] Z. Jing, D. Goldgof, and R. Kasturi, "A new edge-based text verification approach for video," in *Pattern Recognition, 2008. ICPR 2008. 19th International Conference on*, 2008, pp. 1-4.

[4] P. Yi-Feng, H. Xinwen, and L. Cheng-Lin, "A Robust System to Detect and Localize Texts in Natural Scene Images," in *Document Analysis Systems, 2008. DAS '08. The Eighth IAPR International Workshop on*, 2008, pp. 35-42.

[5] D. Chen, J. M. Odobez, and H. Bourlard, "Text detection and recognition in images and video frames," in *Pattern Recognition*. vol. 37: Elsevier, 2004, pp. 595-608.

[6] K. I. Kim, K. Jung, and J. H. Kim, "Texture-based approach for text detection in images using support vector machines and continuously adaptive mean shift algorithm," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*. vol. 25, 2003, pp. 1631-1639.

[7] Q. Ye, Q. Huang, W. Gao, and D. Zhao, "Fast and robust text detection in images and video frames," in *Image and Vision Computing*. vol. 23: Elsevier, 2005, pp. 565-576.

[8] C. Xiangrong, C. Xiangrong, and A. L. Yuille, "Detecting and reading text in natural scenes," in Computer Vision and Pattern Recognition, 2004, pp. II-366-II-373 Vol.2.

[9] W. Xiufei, H. Lei, and L. Changping, "A New Block Partitioned Text Feature for Text Verification," in *Document Analysis and Recognition, 10th International Conference on*, 2009, pp. 366-370.

[10] L.Breiman. Random forests. *Machine Learning*,45:5-32,2001

[11] A. Bosch, A. Zisserman, and X. Muoz, "Image Classification using Random Forests and Ferns," in *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*, 2007, pp. 1-8.

[12] J.Winn and A. Criminisi. Object class recognition at a glance. *CVPR*,2006.

[13] C. Jung, Q. Liu, and J. Kim, "Accurate text localization in images based on SVM output scores," *Image and Vision Computing,* vol. 27, pp. 1295-1301, 2009.

[14] L. Huiping, D. Doermann, and O. Kia, "Automatic text detection and tracking in digital video," *Image Processing, IEEE Transactions on,* vol. 9, pp. 147-156, 2000.

[15] S. M. Hanif, L. Prevost, and P. A. Negri, "A cascade detector for text detection in natural scene images," in *Pattern Recognition, 2008. ICPR 2008. 19th International Conference on*, 2008, pp. 1-4.

[16] M. Anthimopoulos, B. Gatos, and I. Pratikakis, "A two-stage scheme for text detection in video images," *Image and Vision Computing,* vol. 28, pp. 1413-1426.