Scale-Based Local Feature Selection for Scene Text Recognition

Boyu Zhang, Jia Feng Liu, Xiang Long Tang
School of Computer Science and Technology,
Harbin Institute of Technology,
Harbin 150001, China

***Abstract*—Scene text recognition has drawn increasing concerns from the OCR community in recent years. Among numerous methods that have been proposed, local feature based methods represented by bag-of-features (BoFs) show notable robustness and efficiency. However, as the existing detectors are based on assumptions about local saliency, a vast number of non-informative local features would be detected in the feature detection stage. In this paper, we propose to remove non-informative local features by integrating feature scales with stroke width information. Experiments taken both on synthetic data and real scene data show that the proposed feature selection method could effectively filter non-informative features and improve the recognition accuracy.**

***Keywords*—Scene Text Recognition; Local Feature; Stroke Width**

I. INTRODUCTION

In recent years, scene text recognition (STR) [1] technologies have got increasing concerns from OCR community and other related fields. Compared with surrounding text, scene text is more connected to image contents in most cases. Thus the rich semantic information contained in scene text often plays vital roles in a host of computer vision applications, including impaired people assist, visual land-mark robot navigation and intelligent traffic system.

Even numerous potential applications exist, the STR is still challenging due to the following disadvantages: (1) The scales of scene text, even in same sentences, vary a lot; (2) The shapes and styles may be different since scene text are specially designed to fit different requirements; (3) Scene images always contain illumination changes, viewpoints variations and other disadvantages such as a non-flatness surface; (4) In most cases, no context information is provided.

During the past decades, a number of methods are proposed in response to these disadvantages. The existing methods inSTR area could be divided into two categories according toothier basic ideas. One of which is to achieve accurate STR by developing traditional OCR methods. Most approaches under this idea contain three procedures, which are, text detection, segmentation and character recognition. For instance, Chenand Yuille [2] train strong classifier which contains multiple features by integrating weak classifiers with AdaBoost to extract text regions, then text are recognized by employing commercial OCR software. Coates et al. [3] apply scalable leaning algorithm to feature extraction, text detector and classifier to produce high accurate STR system. Kai et al. [4]designed an end-to-end system for scene text recognition, in which Random Fern [5] is utilized as raw character detector as well as classifier. Moreover, they proposed to improve the accuracy of STR by introducing pre-defined vocabulary.

Another idea is to treat scene text as objects. Thus researchers can transplant object recognition methods that are proposed mostly against image degradations and uncontrolled environments into STR area. For example, De Campus et al.[6] build up a STR framework by following classic BoFsmethods in which sample images are described by frequency histogram of local features. They also compare the effectiveness of different local descriptors by taking experiments one representative benchmark. Zheng et.al [7] recognize scene characters by matching detected SIFT [8] features between input samples and pre-build template images. Different from BoFs method that totally omits position information, they consider the relative position of local features by using MPLSH[9]. Diem and Sablatnig [10] build a historical document analysis system based on local descriptors and achieve a state-of-art accuracy for ancient character recognition.

Among these methods, the ones based on local features [6],[7], [10] show notable robustness and effectiveness, especially when in small sample size situations and situations containing image degradations [11]. They are more robust because they represent sample images using sets of local features and omitting other highly variable factors. It is obvious that their accuracy largely depends on the effectiveness of detected local features. However, even most local feature detectors assume that salient image patches are informative, the meanings of effective are different in different applications. Specific to our problem, not all detected saliency image patches reflect local structures of characters. Thus, for improving the accuracy, criteria are needed to filter features which are not related to the text.

In this paper, we focus on local feature based STR and propose a novel criterion which integrate stroke width information with local feature scales to remove non-informative local features and achieve higher accuracy. Our idea is based on the fact that text is constituted by strokes with specific width. Thus there should be an appropriate proportion between local feature scale and the corresponding stroke width if these features reflect local text structures such as corner and cross. Experiments taken on both natural and synthetic text images show that the proposed approach could effectively improve the accuracy of local feature based STR.

II. Related Works

Many techniques are developed for filtering redundancies and noises from original features set. In this paper, we make the specific consideration about methods based on codebook model. A classical codebook method includes local feature detection, codebook generation, quantization, and finally classification. Most efforts for feature selection are taken on codebook generation stage and code-word selection stages. In this section, we briefly introduce typical existing methods according to their categories and discuss differences between these methods and proposed method in the end.

A. Compact Codebook Generation

In codebook generation stage, the algorithm seeks for a group of code words (also referred as 'codebook'), which could describe the feature space effectively. A vast number of methods are proposed to generate effective codebook. For instance, Tuytelaars and Schmid [12] extract high-dimensional descriptors for sample images by partitioning feature space using lattices with regular sizes and then combine similar dimensions to make the descriptors more compact. The most widely applied idea is to get codebook utilizing unsurprised cluster algorithms such as K-means [13], which get the most descriptive k centers by minimizing the variance between k centers and the training data. Different from k-means that is dense sensitive, Jurie and Triggs [14] proposed a radius-based clustering which clusters all features within a fixed radius of similarity radius to one cluster.

B. Code-word Selection

Besides generating a compact codebook, a host of algorithms are proposed for picking the most effective subset from the original codebook. Code-word selection is equal to feature selection problem since sample images are rep- resented by frequency histograms of code-words and each bin corresponding to a feature dimension. Distinguishing by whether class labels are given existing methods could be divided into supervised and unsupervised ones.

Supervised methods analyze the relationship between the class labels and code words and then pick more discriminate subset based on pre-defined criteria. Literature [14] gives a performance evaluation for three typical methods including MI [15], OR [16] and Linear SVM weights [17] on representative datasets. Moosmann et al. [18] proposed to build supervised indexing trees using an ERC-Forest that considers semantic labels as stopping tests. The work in [19] aims to find the Descriptive Visual Words (DVWs) and Descriptive Visual Phrases (DVPs) for each image category.

For unsupervised situations, Zhang et al. [20] proposed to pick out the most discriminative code words which lead to minimal fitting errors between data matrix and indicator matrix. Maximum variance selects features with the largest variances and unsupervised feature selection for PCA selects a subset of features that can best reconstruct other features. Laplacian score [21] selects features tat preserve the local geometrical structure best. Q-α [22] measures the cluster coherence by analyzing the spectral properties of the affinity matrix.

C. Proposed Method

Different from the above methods, the proposed method in this paper filters non-informative features by per- forming a pre-selection based on analyzing both feature scale and stroke width information. Its advantage is that the algorithm effects before codebook generation stage and thus could avoid errors that occur in the following process. This means the proposed methods could be more effective when facing small sample size problems, which are common in STR and historical document analysis.

III. Scale-Based Local Feature Selection

The fundamental assumption of designing most local feature detectors is that salient image patches are informative. In fact, the concepts of 'informative' are different in different situations. Specifically, in STR process, it is not promised each salient patches indeed reflects character structure. Thus criteria are needed to remove features that are not effective.

According to whether they are helpful for distinguishing different characters, we divide detected local features into informative and non-informative. Features belong to the first category always localize in character bounding-boxes and they are salient since they contain character structures such as corners and stroke crosses. In contrast, most features that belong to the second category are generated by cluttered background and noises, thus do not provide information forSTR. It is worthwhile to emphasize that large local features that cover the majority of a character should be categorized into the second type since these features are not robust enough when numerous variations are included.

However, it is difficult to remove non-informative local features automatically as it is difficult to give a formally definition for non-informative features. The target can be achieved by training a binary classifier that could distinguish on-informative features from informative ones, however, a large number of training samples are needed to train such a classifier and the existence of varies fonts makes sample collecting rather difficult. Moreover, labeling all features manually is labor expensive and hardly objective. Another idea is to optimize learned codebook according to class label as we discussed in section II, which is under sophisticated mathematical model. These methods that select features by analyzing the relationship between code words and class labels also need large training dataset.

In this paper, we propose a novel local feature selection criterion that selects effective local features based on the ratio between character stroke width and local feature scale.

A. Feature Scale and Stroke Width

Our idea is based on the observation that it is impossible to write small character with wide strokes and large characters with thin strokes. Thus the ratio between character size s_c and stroke width w in the text area should keep within a reasonable range to ensure the character is recognizable. At the same time, for each detected local feature which reflects a local structure on character, its scale s_f should also be indirect

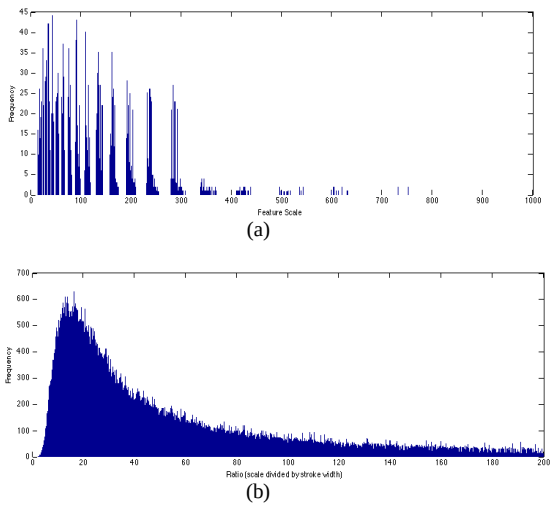


Fig. 1. Diagram (a) shows the frequency histogram of feature scales local features extracted form 'char 74k' dataset. (b) is the frequency histogram of ratio parameters that is calculate by feature scales divided by corresponding stroke width.

Proportion to character scale s_c abided by commonsense. This means that for a reasonable character, the scale of a representative local feature should have a stable ratio r with stroke width w.Based on this idea, we can filter non-effective features by checking whether the ratio r is in an interval $[\mathrm{r}_{\min}, \mathrm{r}_{\max}]$.

The reason we do not directly apply character size for feature selection is that local structures are directly instituted by strokes and thus the ratio between stroke width and feature scale is more stable than the ratio between character size and feature scale. Moreover, stroke width is more accurate then character size in two reasons. Firstly, the segmentation in scene images is difficult which would lead to inaccurate character size. Secondly, characters in the same size have different stroke width because of the existence of multi-font.

To prove this, we count the frequency histograms of the detected local features according to their feature scales and ratio parameters respectively. The definition of stroke width and the calculation of ratio parameters are described in detail in section IV. Fig 1(a) shows the frequency of local feature scale and Fig 1(b) gives the frequency of the ratio between feature scale and corresponding stroke width. We find that the ratio parameter depends on a uniform long-tail distribution which certify that a relationship exists between local feature scales and stroke width.

B. Scale-based Local Feature Selection

Typical local feature detectors such as SIFT and Multi-Scale Harris contain three stages. In the first stage, for each pixel $I(\mathrm{i},\mathrm{j})$ in an image I, its local saliency H corresponding to scale s is evaluated by using measurement function F. By noting the neighborhood of point $I(\mathrm{i},\mathrm{j})$ as $r(\mathrm{i},\mathrm{j})$, we have:

$$H(i,j,s) = F(\mathrm{r}(i,j,s)) \tag{1}$$

Then the algorithm searches local extreme through both spatial and scale space to find local maximums as candidate feature points, which we note as C. At last, a global thresholding process is taken on C abide by following equation:

$$L_{i,j} = \begin{cases} 1, if\ \ H(i,j,s) > th_s \\ 0, else \end{cases} \tag{2}$$

Where $L_{i,j}$ indictors whether pixel $r(\mathrm{i},\mathrm{j})$ is the center of an acceptable local feature and th_s is the threshold of feature saliency. Different from the above process considering the local saliency only, in our work, the relationship between the feature scale s and the stroke width w is also considered. Thus the probability that a local region is effective could be described as $P(H,\mathrm{s},\mathrm{w})$. According to Bayes formula, we have

$$P(H,\mathrm{s},\mathrm{w}) = P(H \mid \mathrm{s},\mathrm{w})P(\mathrm{s},\mathrm{w}) \tag{3}$$

Noticing that the calculation of local saliency H is independent to stroke width w, the probability $P(H,\mathrm{s},\mathrm{w})$ could be simplified into $P(H \mid s)$. Furthermore, in this paper, we describe the relationship $P(\mathrm{s},\mathrm{w})$ between s and w by a sign function of ratio r and use another sign function to describe $P(H \mid s)$, we get

$$L_{i,j} = P(H)\,\mathrm{P}(\mathrm{r}_{i,j}) \tag{4}$$

where

$$P(r) = \begin{cases} 1, r \in [r_{\min}, r_{\max}] \\ 0, else \end{cases} \tag{5}$$

and

$$P(H) = \begin{cases} 1, H > th_s \\ 0, \mathrm{else} \end{cases} \tag{6}$$

Thus we could give the feature selection algorithm based on the above analysis. According to Algorithm 1, we can improve the accuracy and efficiency by removing non-informative local features. Section IV demonstrates the effect of the proposed algorithm.

Algorithm 1 Scale-based Local Feature Selection

(a) (b) (c)

Fig. 2. (a) and (b) shows character samples from Fnt and NS data, respectively. (c) shows Chinese word samples form CH data

IV. Using The Template

In this section, we verify the effect of the proposed scale-based feature selection algorithm with experiments on representative benchmarks. Section IV demonstrate the effect of the proposed algorithm.

A. Experiment Setup

*1) Experimental Data:*To prove that local features with proper scales are more effective, we conduct experiments on a representative benchmark which is referred as 'char74k' [6]. The 'char 74k' dataset contains both synthetic and natural samples. Synthetic samples include 52 classes of English characters (capital letters and lower case letters) and 10 classes of numbers (0~9). For each class, 1016 character samples are generated according to 256 different system fonts with 4 different styles. For natural samples, characters are cropped manually from scene images. Fig 2(a) and Fig 2(b) shows some typical samples of 'Fnt' data and 'NS' data in this benchmark. This dataset is selected for two reasons. Firstly, it contains typical scene character samples which are segmented manually and labeled in detail. Secondly, synthetic data could be used as baseline in our experiment since these samples certify accurate stroke width information and all detected local features are useful for character recognition. Moreover, we collect our own Chinese words dataset (the dataset will be referred as 'CH' in the following parts of this paper) beside the above benchmark using Internet searching engine according to 12 different key words. For each text image we get, accurate text regions are cropped and labeled manually. Examples of CH data are shown in Fig 2(c).

*2) Local Feature Detection:*We employ two typical detectors, which are, Hessian-affine and difference of Gaussian (DoG). According to the literature [6], the combination of DoG detector and SIFT descriptor performs much better than others.

*3) Stroke Width Extraction:*In this paper, stroke width information is extracted by utilizing stroke width transform [23]. For each pixel in a text image, if it is localized between two edges pixels with opposite gradient directions, its stroke width value is defined as the distance between these two edge pixels. If more than one pair of edge pixels are found, the stroke width value is set as the minimum one. On the contrary, stroke width value is set as infinite when the algorithm cannot find pixels like that. For more details about stroke width extraction, readers could refer to the original paper by Epstein et.al.[23]. Two factors should be considered for extracting precise stroke width. The first one is the thresholds for edge detector (Canny here) should be selected very carefully since the precision of SWT heavily depends on the results of edge detection. The second is that the algorithm needs to know whether the character pixels are darker than the background or opposite. In practice, it is without any difficulties to assign parameters of edge detector for synthetic data as these images have high contrast (binary images, actually). Moreover, all synthetic samples have darker pixels compared to the background. For natural images, thresholds of Canny operator are assigned much lower by considering the image contrast and the contrast between text and background are assigned manually.

Based on detected local features and extracted stroke width value, we can calculate the ratio r for each local feature.

B. Character and Word Recognition

Text recognition is achieved based on classic bag-of-features framework, which is similar to literature [6]. In our experiments, 30 training samples and 15 testing samples are selected randomly for each class. Then local features are detected and described as mentioned above. The visual word vocabulary is generated by using k-means cluster algorithm, and the number P of visual words for each class is assigned equally (varies from 2 to 10 in the following experiments).

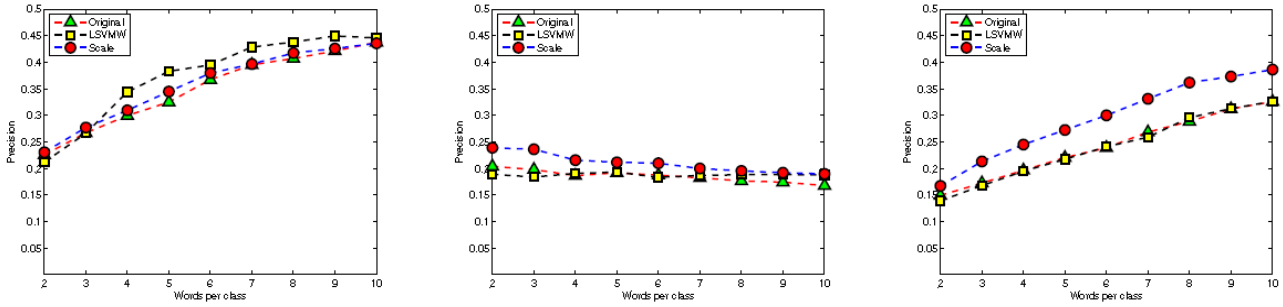
Finally, each sample is quantized into feature vector according to the vocabulary and thus each sample image is described bya $P \times C$ dimension vector where C is the number of classes. Support vector machine (SVM) with RBF kernel is chosen as classifier due to its effectiveness and representativeness and '1 VS all 'strategy is employed to solve multi-class problem.

Besides, we perform recognition separately for numbers, lowercase letters and capital letters to avoid the influence of similar symbols such as 'o'and '0', 'p' and 'P'. Thus the accuracy for NS and Fnt data is calculated by using the weighted average according the following equation

$$Acc = \frac{C_c}{C} Acc_c + \frac{C_l}{C} Acc_l + \frac{C_n}{C} Acc_n$$

where C_c, C_l and C_n is the class number of capital letters, lowercase letters and numbers and $C = C_c + C_l + C_n$.

In the feature selection stage, a group of samples for each class are selected to find the best threshold for filtering especially large or small features. For each training process, we



(a)(b) (c)

Fig. 3. Diagram (a) shows the recognition accuracy of LSVMW, proposed method and original BoFs based method on Fnt data. Diagram (b) and (c) shows the corresponding results on NS and CH data, respectively.

TABLE I. IMPROVEMENTS BROUGHT BY SCALE-BASED LOCAL FEATURE SELECTION

Words per Class		*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*	*10*
Rate of Improvement (%)	**Fnt**	2.20	3.84	3.27	6.13	3.39	0.39	2.51	1.07	-0.42
	NS	17.17	19.34	15.61	10.31	11.94	9.61	10.59	10.19	13.04
	CH	12.05	23.61	24.70	23.64	25.56	23.43	25.31	19.58	18.57

remove features that have extremely large or small ratio parameter as percentage. The best filter threshold is found by employing grid search. For Fnt data, the algorithm search the best threshold from 1% to 10% for both large and small sides. The reason for limiting the searching range is that very few non-informative features are detected for Fnt data. The experimental results also show that the best thresholds in the neighborhood of 1% in most cases for Fnt data. We can find that the selection slightly improve the accuracy of Fnt data. Besides, the results of feature selection using linear SVM weight is also shown in the Fig 3. The results of MI and IG are not attached as LSVMW over-performs them. We can discover that both Scale-based feature selection and LSVMW-based method can improve the accuracy of Fntdata. However, the improvement of scale-based method is not very obvious and weaker then LSVMW-based one. The reason is that most detected local features are informative since no cluster background and noises are included in Fnt data. For the NS data which include more noises, scale-based feature selection strategy overruns both original data and LSVMW-based feature selection. The results show that the scale-based feature selection brings more benefits when a rare word number is used and the efforts of LSVMW is close to our method when the number of words increases. The reason is that when rare word number is used, the influence of noises is more obvious in that error code words will reduce the accuracy, and the proposed method is more effective for filtering non-informative features and avoiding the generation of error code words.

The accuracy for CH data is calculated under the same method. We can discover that the proposed scale-based feature selection method obviously overruns original and LSVMW-based method. This encouraging result further proves that we can filter non-informative local features by considering both feature scales and stroke width. To examine the improvement of the proposed method in greater detail, the recognition accuracy of original data and filtered data is calculated. Moreover, the improvement brought by stroke width information is evaluated as follows: noting the recognition accuracy on original feature set as C_{ori} and C_{sel} as accuracy on selected feature set, the improvement C_{imp} can be evaluated as $C_{imp} = (C_{sel} - C_{ori}) / C_{ori}$. The results are shown in Table I. From Table I, we can see that supervised feature selection algorithms such as LSVMW are more effective for clean data and the proposed method is more effective when samples contain more noises and degradation such as NS data and CHdata.

V. CONCLUSION

In this paper, we proposed a new approach for filtering text-independent local features by considering both stroke width information and feature scales. The proposed approach is tested on representative benchmarks and the encouraging experimental results (a maximum improvement of 25.56% for CH data and 19.34% for natural data) prove the existence of relevancy between stroke width and feature scales. Different from traditional methods which need a group of training data, the proposed approach can effectively filter on-informative local features when only a few samples are used. Moreover, it is notable that the proposed approach is evidently effective for degraded images and small sample size situations. These two advantages ensure the proposed method could be widely applied in the fields such as historical document analysis and text-associate image retrieval. At the same time, we can find that there is much room for improvement in recognition rate for local feature based algorithms. Therefore, our future work include developing probability model which aims at increasing the accuracy of local feature based STR and building end-to-end scene text analysis system.

ACKNOWLEDGMENT

This work was supported by National Natural Science Foundation of China (grant No.61073128).

REFERENCES

[1] K. Jung, K. In Kim, and A. K Jain, "Text information extraction in images and video: a survey," Pattern recognition, vol. 37, no. 5, pp. 977-997, 2004.

[2] X. Chen and A. L. Yuille, "Detecting and reading text in natural scenes," in Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on, vol. 2. IEEE, 2004, pp. II-366.

[3] A. Coates, B. Carpenter, C. Case, S. Satheesh, B. Suresh, T. Wang, D. J. Wu, and A. Y. Ng, "Text detection and character recognition in scene images with unsupervised feature learning," in Document Analysis and Recognition (ICDAR), 2011 International Conference on. IEEE, 2011, pp. 440-445.

[4] K. Wang, B. Babenko, and S. Belongie, "End-to-end scene text recognition," in Computer Vision (ICCV), 2011 IEEE International Conference on. IEEE, 2011, pp. 1457-1464.

[5] 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. IEEE, 2007, pp. 1–8.

[6] T. de Campos, B. R. Babu, and M. Varma, "Character recognition innatural images," 2009.

[7] Q. Zheng, K. Chen, Y. Zhou, C. Gu, and H. Guan, "Text localization and recognition in complex scenes using local features," in Computer Vision–ACCV 2010. Springer, 2011, pp. 121-132.

[8] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," International journal of computer vision, vol. 60, no. 2, pp. 91-110, 2004.

[9] Q. Lv, W. Josephson, Z. Wang, M. Charikar, and K. Li, "Multiprobe lsh: efficient indexing for high-dimensional similarity search," in Proceedings of the 33rd international conference on Very large databases. VLDB Endowment, 2007, pp. 950–961.

[10] M. Diem and R. Sablatnig, "Are characters objects?" in Frontiers inHandwriting Recognition (ICFHR), 2010 International Conference on. IEEE, 2010, pp. 565-570.

[11] K. Das and Z. Nenadic, "An efficient discriminant-based solution for small sample size problem," Pattern Recognition, vol. 42, no. 5, pp. 857-866, 2009.

[12] T. Tuytelaars and C. Schmid, "Vector quantizing feature space with a regular lattice," in Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on. IEEE, 2007, pp. 1-8.

[13] D. Lee, S. Baek, and K. Sung, "Modified k-means algorithm for vectorquantizer design," Signal Processing Letters, IEEE, vol. 4, no. 1, pp.2-4, 1997.

[14] F. Jurie and B. Triggs, "Creating efficient codebooks for visual recognition,"in Computer Vision, 2005. ICCV 2005. Tenth IEEE InternationalConference on, vol. 1. IEEE, 2005, pp. 604-610.

[15] M. Vidal-Naquet and S. Ullman, "Object recognition with informativefeatures and linear classification." in ICCV, vol. 3, 2003, p. 281.

[16] J. Brank, M. Grobelnik, N. Milic-Frayling, and D. Mladenic, "Interactionof feature selection methods and linear classification models," inWorkshop on Text Learning held at ICML, 2002.

[17] Y. W. Chang and C. J. Lin, "Feature ranking using linear svm," Causationand Prediction Challenge Challenges in Machine Learning, Volume2, p. 47, 2008.

[18] F. Moosmann, B. Triggs, F. Jurie et al., "Fast discriminative visualcodebooks using randomized clustering forests," Advances in NeuralInformation Processing Systems 19, pp. 985-992, 2007.

[19] S. Zhang, Q. Tian, G. Hua, Q. Huang, and W. Gao, "Generatingdescriptive visual words and visual phrases for large-scale imageapplications," Image Processing, IEEE Transactions on, vol. 20, no. 9,pp. 2664-2677, 2011.

[20] L. Zhang, C. Chen, J. Bu, Z. Chen, S. Tan, and X. He, "Discriminativecodeword selection for image representation," in Proceedings of theinternational conference on Multimedia. ACM, 2010, pp. 173-182.

[21] X. He, D. Cai, and P. Niyogi, "Laplacian score for feature selection," inAdvances in neural information processing systems, 2005, pp. 507-514.

[22] L. Wolf and A. Shashua, "Feature selection for unsupervised andsupervised inference: The emergence of sparsity in a weight-basedapproach," The Journal of Machine Learning Research, vol. 6, pp.1855-1887, 2005.

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