2013 12th International Conference on Document Analysis and Recognition

Automatic Labeling for Scene Text Database

Masakazu Iwamura, Masaki Tsukada and Koichi Kise
Graduate School of Engineering, Osaka Prefecture University
Email: {masa, kise}@cs.osakafu-u.ac.jp

***Abstract*—It is thought that a large quantity of data improve quality of recognition. A large database, however, is not easy to obtain. The hardest task is labeling (also known as ground truthing), which usually requires human intervention. Since labeling by human is laborious and costly, labeling without human (automatic labeling) or minimization of human intervention (semi-automatic labeling) are ideal scenarios. As a step toward realization of the scenarios, knowing how much an automatic labeling system can perform without human intervention is important. In the current paper we propose a comprehensive automatic labeling technique for a scene text database, which performs segmentation and labeling for unsegmented and unlabeled character images. To our best knowledge, this is the first method to realize the comprehensive process for automatic labeling for scene text databases In experiments, we confirmed that the proposed method could add new unlabeled data in parallel with improving recognition performance of the classifier.**

I. Introduction

Collecting larger data is directly connected to better performance in pattern recognition. A simple classifier such as k-nearest neighbor classifier with large data outperforms a sophisticated classifier such as support vector machine with less data. In this sense, quantity of data improves quality of recognition. Therefore large databases are always demanded for better recognition performance.

In spite of the fact that getting large data is getting easy, a large database is still not easy to obtain. Its bottleneck is labeling (also known as ground truthing), which usually requires human intervention. Since labeling by human is laborious and costly, labeling without human (automatic labeling) or minimization of human intervention (semi-automatic labeling) are ideal scenarios. As a step toward realization of the scenarios, knowing how much an automatic labeling system can perform without human intervention is important.

In this paper, we focus on automatic labeling for scene text databases. Figure 1 shows an overview of the proposed system, which consists of *segmentation by recognition* and *labeling by recognition* processes. As the labeling process, we applied the self-corrective recognition approach [1] for segmented character images proposed in [2]. It improves the recognition performance of the classifier with unlabeled data by a process called *re-training*. In order to complete the whole process, we propose a method realizing the segmentation process. By combining the proposed segmentation method and an improved version of the labeling method [2], we realize a comprehensive automatic labeling technique for a scene text database. To our best knowledge, this is the first method to realize the comprehensive process for automatic labeling for scene text databases.

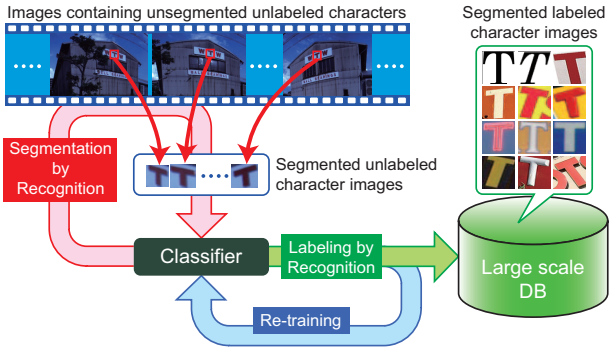


Fig. 1. An overview of the proposed comprehensive system for automatic labeling of scene texts. After initial training of a classifier with labeled data, *segmentation by recognition* is applied to *images containing unsegmented unlabeled characters*. Then, *labeling by recognition* is applied to *segmented unlabeled character images*. They are stored in the database with the predicted labels as *segmented labeled character images* and also used for re-training of the classifier.

II. Related Work

A. Segmentation

Existing segmentation methods for scene texts are categorized into some groups. One is texture-based approach, which uses texture information (e.g., [3]). Another is region-based approach, which finds connected components or blobs (e.g., [4]–[7]). There is their hybrid approach [8]. In this paper, yet another approach based on SIFT like local feature (e.g., [9], [10]) is employed. Unlike other approaches, it does not simply segment characters but also recognize them simultaneously. In this sense, this approach is regraded as *segmentation by recognition*. We employ a method proposed by ourselves [10]. However, it is hard to extract a sufficient number of local features from low resolution or blurred images. Thus we apply the dense sampling approach which use grid points as feature points.

B. Labeling

Recently automatic and semi-automatic labeling methods attract researchers. Automatic labeling realizes labeling without human. A representative approach is semi-supervised learning [11]. The self-corrective recognition algorithm [1], [2] is regarded as self-training in the context of semi-supervised learning, which is the most simple approach among exiting approaches. In handwriting recognition, a series of papers using the self-training and co-training (which is better than self-training) are published by the same group (e.g., [12], [13]). Semi-automatic labeling minimize human intervention. A representative approach is active learning (e.g., [14]), which asks a small number of questions to human.

1520-5363/13 $26.00 © 2013 IEEE
DOI 10.1109/ICDAR.2013.276

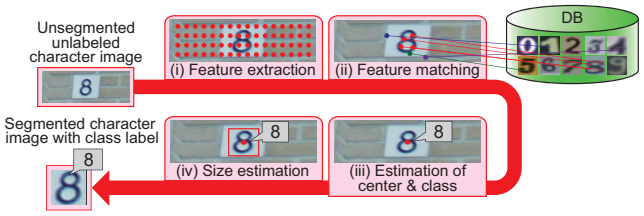


Fig. 2. An overview of the proposed segmentation method, which consists of (i) Feature extraction, (ii) Feature matching, (iii) Estimation of the center and class label of a character region, and (iv) Size estimation.

III. Proposed Method

Intuitively the proposed method is a mixture of two methods. One is a scene character recognition method based on arrangement of local features and voting technique [10]. The other is a self-corrective recognition method based on an approximate nearest neighbor classifier [2]. Since both techniques use approximate nearest neighbor search (ANNS), they can be unified with minimum modification. The detail of the proposed method is presented below.

A. Initial training

In the initial training, feature vectors extracted from segmented and labeled images are stored in the database. We call the images *reference images*. In this paper, the following processes were performed. The images were normalized to 96×96 pixels and dense sampling was performed to determine 200 feature points consisting of three scales (10×10 grid for the scale 2, 8×8 for 4 and 6×6 for 6). Then, PCA-SIFT descriptor [15] was used to obtain feature vectors.

B. Segmentation by recognition

Figure 2 shows an overview of the proposed segmentation method. The following process is applied to unsegmented images.

(i) Feature extraction process is similar to the one in the initial training except normalization and selection of grid used for dense sampling. A given image is adaptively normalized according to the image size. Letting n be an integer larger than 4, the image is magnified by n horizontally and $n/2$ vertically (e.g., horizontally 5 times and vertically 2.5 times) to n is determined so that the height will be larger than 200 pixels. One exception is that if the original image height is larger than 200 pixels, $n = 2$ is used. Then feature points are determined in every 4 pixels.

(ii) Feature matching process is performed using an ANNS method [16]. The feature with the smallest distance from a query feature is searched in the database. The features are regarded as corresponding if they satisfy

$$\frac{d_{\text{nn}}}{d_{\text{2nn}}} < t_d, \tag{1}$$

where d_{nn} and d_{2nn} represent the distances to the nearest and second nearest features of the reference images, respectively, and t_d is a threshold. The reason for filtering out features is that we found wrong correspondences decrease as the ratio becomes small.

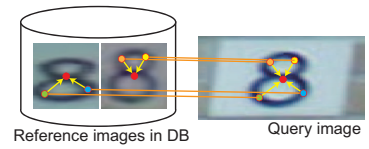


Fig. 3. (iii) Estimation of the center and class label of a character region.

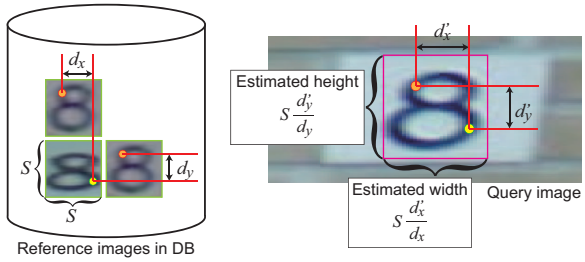


Fig. 4. (iv) Size estimation.

(iii) Figure 3 illustrates an overview of estimation of character center and class with reference point (RP) [17], which consists of (1) Storing the relative positions from the feature points to the centers of the reference images, (2) Projection of the relative positions onto the query image, and (3) Estimation of the center of a character region. In the process (3), the center of a character region is estimated as follows. First, RPs corresponding to a class are found in the query image. If the number of the RPs within a distance d_c from one of the RPs is larger than n_c, a cluster containing them is created. Then, the medians in x and y-coordinates of the RPs in the cluster are determined as the coordinates of the character center. Finally, the class label of the character is determined as that of the RPs.

(iv) Figure 4 illustrates an overview of size estimation. For each pair of two correspondences of feature points, distances in x and y-coordinates are measured in both query and reference images. Let d_x and d_y be the distances in x and y-coordinates in the reference images. Let d'_x and d'_y be those in the query image. Letting S be the size of the normalized reference images, the width and height of the character in the query image are estimated as $S \cdot d'_x/d_x$ and $S \cdot d'_y/d_y$, respectively. The procedure is applied to all the pairs. Finally, the width and height of the character are determined as their medians.

After the procedure above finishes, if there are regions overlapping each other with the same class label, the character region is redefined by the smallest rectangle including the regions.

C. Labeling by recognition

Figure 5 shows an overview of the proposed labeling method. The following process is applied to segmented unlabeled images.

(I) Initial training is already presented above since the database is shared with the segmentation and labeling methods.

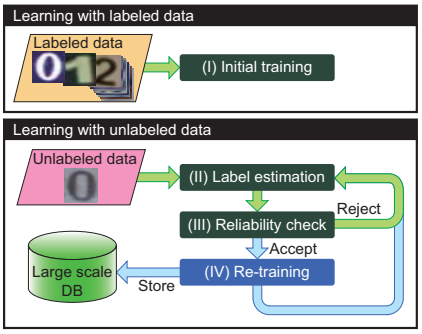


Fig. 5. An overview of the proposed automatic labeling method, which consists of (I) Initial training, (II) Label estimation, (III) Reliability check, and (IV) Re-training.

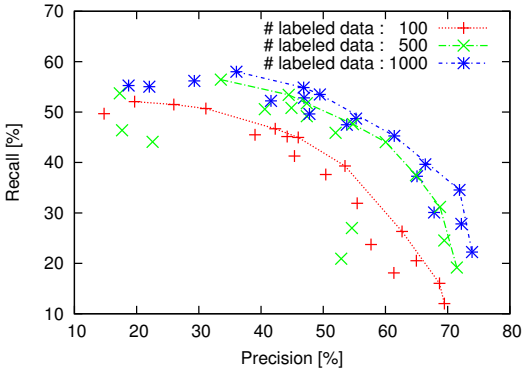


Fig. 6. Exp 1: Recall-precision curve for segmentation obtained by changing d_c and n_c. The points connected with lines are Pareto optimal solutions.

(II) For label estimation, feature extraction and feature matching are performed in the same manner as the segmentation method. Then, votes are cast for the class labels of the nearest features with a weight of $1/N_i$, where N_i is the number of features of i-th class stored in the database. The weight is required so that the expected number of feature vectors is empirically equalized with the weight; more feature vectors are expected to appear in the query image if more feature vectors are extracted from the reference images. Then, the class label is estimated as the one having the highest votes.

(III) Reliability check is performed to avoid failure in image acquisition and re-training. Let s_1 and s_2 be the highest and second highest votes, respectively. Then, $c = s_1/s_2$ is defined as confidence, which is no less than 1. The condition for the confidence c is defined by

$$t_l < c < t_u, \tag{2}$$

where t_u and t_l are the upper and lower bounds for the confidence. The reason that the upper bound for the confidence exists is that we observed the cases where only one class has many votes and others do not when a part of the background is wrongly segmented and recognized as a character. Thus, the upper bound is introduced for filtering out in the cases. If the confidence c satisfies Eq. (2), the estimated label is regarded as reliable. If not, it is rejected.

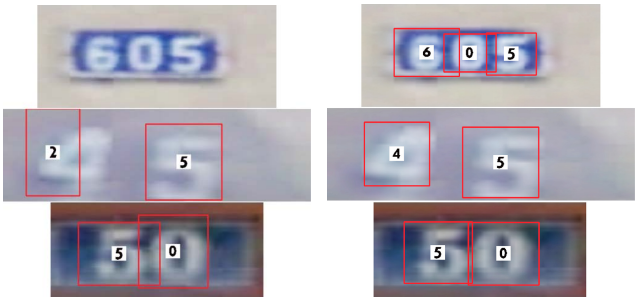
(IV) Re-training is performed only for reliable data. Before the re-training, the same filtering shown in Eq. (1) is performed. This is introduced to avoid reduction in recognition performance of the classifier by storing wrong feature vectors contained in the character region.

IV. Experiments

We used digits images from the *full numbers* format in Street View House Numbers (SVHN) Dataset [18] because they keep the original resolutions and color. They contain three subsets: train, test, extra. We performed three experiments presented below.

A. Experiment 1: Evaluation of segmentation

The proposed segmentation method is evaluated. As the labeled data, 100, 500 and 1,000 images per class were randomly selected from the train subset. As the queries, 1,000



(a) 100 labeled images per class. (b) 1,000 labeled images per class.

Fig. 7. Exp 1: Some of segmentation results.

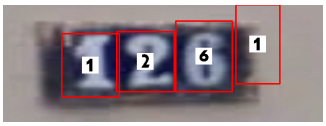


Fig. 8. Exp 1: Typical example of failure case. The plate frame was misrecognized as 1.

images were randomly selected from the extra subset. The thresholds were set to be $t_d = 0.9$, $d_c = 30$ and $n_c = 35$. Evaluation was performed along the way described in [19].

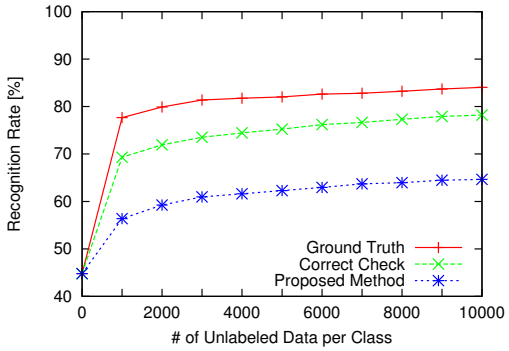
Figure 6 shows the recall-precision curve obtained by changing d_c and n_c. The figure shows that the larger number of labeled data were used, the better recall and precision were obtained. The figure also shows that if the number of labeled data is large, about 60% of recall rate is achieved with about 35% of precision rate. Since high recall is desired for segmentation method, this measure is important. Figure 7 shows some of segmentation results with different number of labeled images. We can observe that characters which were not segmented correctly with fewer labeled images were segmented well with more labeled images. Figure 8 shows a typical example of failure case, where plate frames were misrecognized as 1.

B. Experiment 2: Evaluation of automatic labeling

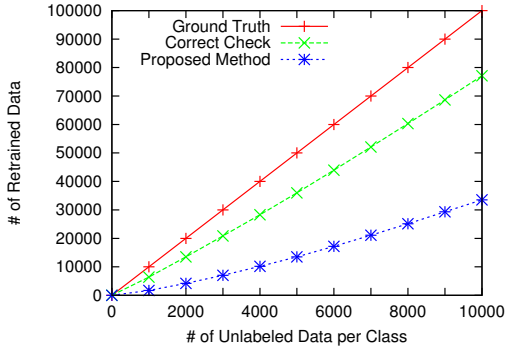
The proposed automatic labeling method is evaluated. Since solo performance of the method is evaluated, only segmented and normalized images were used. Normalization was performed using the character level bounding boxes provided in the SVHN dataset. As the labeled data, 10 images per class were randomly selected from the train subset. As the unlabeled data, 10,000 images per class were randomly selected from the train and extra subsets. As the queries, all the 26,032 images contained in the test subset were used. The thresholds were set to be $t_d = 0.9$ and the lower bound $t_l = 5$. The upper bound t_u was not used (i.e. $t_u = \infty$). Unlabeled data rejected in the reliability check can be used after the classifier is further trained. While this can improve the recognition performance of the classifier, we did not take this way in this paper. That is, the unlabeled data were examined once.

1) Effect on Feature selection: We evaluated the effect on the feature selection shown in Eq. (1), which is also referred as *filtering* above. The feature selection process is carried out twice; one is in the process (ii) of segmentation (referred as *before voting*) and the other is (IV) of automatic labeling (referred as *before re-training*). In the evaluation, each feature selection process was enabled/disabled. Table I shows the result. First, effect on feature selection before voting is examined. (b) and (d) which employ feature selection before voting achieved better *recognition rate after re-training*, compared with (a) and (c) in which all the features were used for voting. The *recognition rates after initial training* of (b) and (d) were less than those of (a) and (c). This was caused by that less number of feature vectors satisfied Eq. (1) in (b) and (d). As a result, we confirmed the importance of selecting features before voting in order to improve the recognition performance. Second, effect on feature selection before re-training is examined. Comparing (c) and (d) which employ feature selection before re-training with (a) and (b) in which all the features were used for re-training, the number of labeled data obtained and accuracy of label estimation were quite different. That is, less unlabeled data were labeled with the feature selection in (c) and (d) while the accuracy was higher. As a result, they achieved better *recognition rate after re-training*, with more reliable feature vectors selected using the selection process. Thorough the experiments, we found that feature selection before both voting and re-training was the best strategy. Thus, we use this strategy hereafter.

2) Effect on the number of unlabeled data: We evaluated the effect on the number of unlabeled data. Figures 9(a) and (b) respectively show the recognition rate and the number of data trained as the number of unlabeled data increases from 1,000 to 10,000 by 1,000. We employed three scenarios. *Ground Truth* represents the scenario that the labels of all the unlabeled data were correctly estimated and they were used for re-training. *Correct Check* represents the scenario that the labels for unlabeled data were estimated by the proposed method but the reliability check was always correct. In this scenario, feature selection before re-training was also applied. *Proposed Method* represents the scenario to employ the proposed method. The primary cause of the difference between Ground Truth and Correct Check was difference of the number of unlabeled data used for re-training. This came from the error of label estimation happens only in Correct Check. Another cause



(a) Relationship between the number of unlabeled data and recognition rate.



(b) Relationship between the number of unlabeled data and the number of data trained (which is equivalent to the number of labeled data obtained).

Fig. 9. Exp 2: Effect on the number of unlabeled data

was that the number of feature vectors stored in the database was less in Correct Check because of feature selection. Since the estimated label was always correct, a larger number of feature vectors improved the recognition rate. The difference between Correct Check and Proposed Method was caused by the difference of the performance of the reliability check since in Proposed Method unlabeled data with wrong labels can be used for re-training. Consequently, the number of unlabeled data correctly used for re-training affected the recognition rate of the classifier after re-training.

3) Effect on the parameters: Figure 10 shows the recall-precision curve obtained by changing the lower bound t_l. The figure shows that the larger number of labeled data were used, the better recall and precision were obtained as in Figure 6. The figure also shows that if the number of labeled data is large, a high precision rate can be achieved. This property is quite important for the automatic labeling method.

C. Experiment 3: Evaluation of unified system

The unified system consisting of the proposed segmentation and automatic labeling methods is evaluated. In the system, the segmentation method is applied to unsegmented unlabeled character images and then the automatic labeling method is applied to the output of the segmentation method. If the class labels estimated in both methods are not consistent, the unlabeled image is rejected.

TABLE I. EXP 2: EFFECT ON FEATURE SELECTION IN AUTOMATIC LABELING. (A) NO FEATURE SELECTION, (B) FEATURE SELECTION BEFORE VOTING, (C) FEATURE SELECTION BEFORE RE-TRAINING, (D) FEATURE SELECTION BEFORE BOTH VOTING AND RE-TRAINING.

	(a)	(b)	(c)	(d)
Recog. rate aft. initial training [%]	46.3	44.8	46.3	44.8
Recog. rate aft. re-training [%]	42.7	50.8	50.3	64.7
# of labeled data obtained	41,360	46,834	30,624	33,531
Accuracy of label estimation [%]	63.7	68.2	81.4	89.0

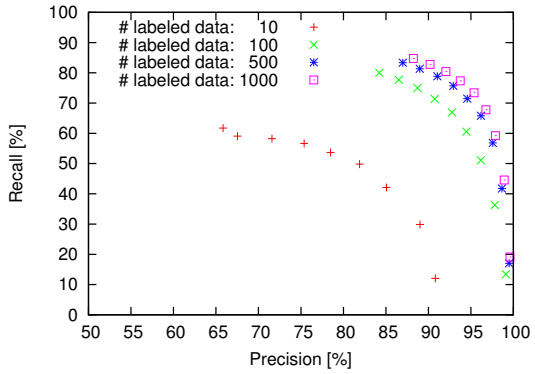


Fig. 10. Exp 2: Recall-precision curve for automatic labeling obtained by changing the lower bound t_l. The upper bound t_u was kept infinity.

As the labeled data, 1,000 images per class were randomly selected from the train subset. As the queries, 5,000 unsegmented images were randomly selected from the extra subset. The total number of characters contained in the images were 13,215. To evaluate the recognition performance of the classifier, all the 26,032 images contained in the test subset were used. The thresholds were set to be $t_d = 0.9$, $d_c = 30$, $n_c = 55$, $t_l = 10/7 \approx 1.43$ and $t_u = 10/3 \approx 3.33$.

Table II shows the experimental result. Comparing *recognition rates after initial training* and *re-training*, about 1.3% was gained through the process. 4,037 unlabeled data were newly collected and stored in the database, out of which 2,947 were correctly labeled. The accuracy of label estimation was 73.0%.

V. CONCLUSION

A large database is not easy to obtain. This is because labeling usually requires human intervention, which is laborious and costly. Thus, labeling without human (automatic labeling) or minimization of human intervention (semi-automatic labeling) are ideal. As a step toward realization of them, in the current paper we proposed a comprehensive automatic labeling technique for a scene text database. The proposed method performed segmentation and labeling for unsegmented and unlabeled character images. To our best knowledge, this is the first method to realize the comprehensive process for automatic labeling for scene text databases Three experiments evaluated the performance of the proposed method. They revealed that the unified system consisting of the proposed segmentation and automatic labeling methods could gain about 1.3% and 4,037 unlabeled data were newly collected and stored in the database, out of which 2,947 were correctly labeled. The accuracy of label estimation was 73.0%. The numbers were not bad for a first trial. In future the system is going to be improved.

TABLE II. EXP 3: RECOGNITION RESULT ON UNIFIED SYSTEM.

Recog. rate aft. initial training [%]	77.5
Recog. rate aft. re-training [%]	78.8
# of labeled data obtained	4,037
Accuracy of label estimation [%]	73.0

ACKNOWLEDGMENT

This work is supported in part by JST CREST project.

REFERENCES

[1] G. Nagy and G. L. Shelton, "Self-corrective character recognition system," *IEEE Trans. Information Theory*, vol. IT-12, pp. 215–222, Apr. 1966.

[2] M. Tsukada, M. Iwamura, and K. Kise, "Expanding recognizable distorted characters using self-corrective recognition," in *Proc. DAS*, 2012, pp. 327–332.

[3] J.-J. Lee, P.-H. Lee, S.-W. Lee, A. L. Yuille, and C. Koch, "Adaboost for text detection in natural scene," in *Proc. ICDAR*, 2011, pp. 429–434.

[4] B. Epshtein, E. Ofek, and Y. Wexler, "Detecting text in natural scenes with stroke width transform," in *Proc. CVPR*, 2010.

[5] P. Sanketi, H. Shen, and J. M. Coughlan, "Localizing blurry and low-resolution text in natural images," in *Proc. IEEE Workshop on Applications of Computer Vision*, 2011.

[6] C. Yao, Z. Tu, and Y. Ma, "Detecting texts of arbitrary orientations in natural images," in *Proc. CVPR*, 2012.

[7] L. Neumann and J. Matas, "Real-time scene text localization and recognition," in *Proc. CVPR*, 2012.

[8] Y.-F. Pan, X. Hou, and C.-L. Liu, "A hybrid approach to detect and localize texts in natural scene images," *IEEE Trans. on Image Processing*, vol. 20, no. 3, pp. 800–813, Mar. 2011.

[9] Q. Zheng, K. Chen, Y. Zhou, C. Gu, and H. Guan, "Text localization and recognition in complex scenes using local features," in *Proc. ACCV*, vol. Part III, 2010, pp. 121–132.

[10] M. Iwamura, T. Kobayashi, and K. Kise, "Recognition of multiple characters in a scene image using arrangement of local features," in *Proc. ICDAR*, 2011, pp. 1409–1413.

[11] X. Zhu and A. B. Goldberg, *Introduction to Semi-Supervised Learning*. Morgan and Claypool Publishers, Sep. 2009.

[12] V. Frinken, A. Fischer, H. Bunke, and A. Fornes, "Co-training for handwritten word recognition," in *Proc. ICDAR*, 2011, pp. 314–318.

[13] V. Frinken, M. Baumgartner, A. Fischer, and H. Bunke, "Semi-supervised learning for cursive handwriting recognition using keyword spotting," in *Proc. ICFHR*, 2012, pp. 49–54.

[14] A. Prakash and D. Parikh, "Attributes for classifier feedback," in *Proc. ECCV*, 2012.

[15] Y. Ke and R. Sukthankar, "Pca-sift: A more distinctive representation for local image descriptors," *Proc. CVPR*, vol. 2, pp. 506–513, 2004.

[16] T. Sato, M. Iwamura, and K. Kise, "Fast and memory efficient approximate nearest neighbor search with distance estimation based on space indexing," IEICE Tech. Rep., PRMU2012-142, Feb. 2013, in Japanese.

[17] M. Klinkigt and K. Kise, "Using a reference point for local configuration of sift-like features for object recognition with serious background clutter," *IPSJ Transactions on Computer Vision and Applications*, vol. 3, pp. 110–121, Dec. 2011.

[18] Y. Netzer, T. Wang, A. Coates, A. Bissacco, B. Wu, and A. Y. Ng, "Reading digits in natural images with unsupervised feature learning," in *Proc. NIPS Workshop on Deep Learning and Unsupervised Feature Learning*, 2011.

[19] C. Wolf and J.-M. Jolion, "Object count/area graphs for the evaluation of object detection and segmentation algorithms," *International Journal on Document Analysis and Recognition*, vol. 8, no. 4, pp. 280–296, 2006.