J. Vis. Commun. Image R. 44 (2017) 21–28

Contents lists available at ScienceDirect

J. Vis. Commun. Image R.

journal homepage: www.elsevier.com/locate/jvci

Store classification using Text-Exemplar-Similarity and Hypotheses-Weighted-CNN ☆

Chao Huang *, Hongliang Li, Wei Li, Qingbo Wu, Linfeng Xu

School of Electronic Engineering, University of Electronic Science and Technology of China, Chengdu 611731, China

ARTICLE INFO

Article history:
Received 24 November 2015
Revised 22 November 2016
Accepted 11 January 2017
Available online 16 January 2017

Keywords:
Store classification
Deep convolutional network
Hypotheses-weighted CNN
Image classification
Text-Exemplar-Similarity

ABSTRACT

Store classification is a challenging task due to the large variation of view, scale, illumination and occlusion. To efficiently distinguish different stores, we introduce two features: Text-Exemplar-Similarity and Hypotheses-Weighted-CNN. For the first feature, the similarity with the discriminative characters is used to represent the text information. For the second feature, we first generate a set of object hypotheses. Then, we introduce two priors: edge boundary and repeatness prior to give a higher weight to the hypotheses enclosing the object. After the generation of two features, a simple and efficient optimization method is used to find the best weight for each feature. Extensive experiments are evaluated to verify the superiority of the proposed method. We built a new 9-class store dataset composed of photos and images from the internet. The experiments show that our method is nearly 10% higher than the state-of-art methods.

1. Introduction

Out-door image classification is a general and practical problem in scene recognition, which is a challenging task due to a large number of complex variation of view, scale and illumination. In out-door scene, store is common and closely related to our daily life. Store classification has great potential for future tasks, such as the application of Google glass and intelligent robot.

The existing methods mainly focus on the recognition of natural scene [1–3] and event [4]. The traditional works follow the Bag-of-Words (BoW) framework consisting of feature extraction, feature coding and classification. The hand-crafted features are employed to capture local information. Then, the coding methods [5,6] are employed to generate the image-level representation, which is fed to classic classifiers [7,8]. Recently, the deep convolutional neural network (CNN) has been widely employed in many image analysis applications, such as object detection [9] and image recognition [1,10]. Compared with the hand-crafted feature, the auto-learnt CNN feature has achieved the breakthrough performance for large-scale image classification task, e.g. ImageNet Large Scale Visual Recognition Challenge [11].

Differing from the natural scene and event, store images have several distinct properties for recognition task. First, there exists a large variation of viewpoint and scale. Second, the category of some stores can be inferred from the text in the signboard.

In this paper, we introduce a store dataset and propose a method to integrate the text information with the image-level feature. Two types of features (e.g. Text-Exemplar-Similarity and Hypotheses-Weighted-CNN) are used for store classification. For the first type of feature, we find the discriminative characters for each category. The similarity scores with these characters are treated as the text-level feature. The extraction of the second feature consists of five steps. The object hypotheses are first generated by selective search. Then, the CNN feature is extracted from each candidate. Meanwhile, objectness score of each proposal is computed based on two priors: "edge boundary" and "repeatness". The signboard and region under signboard are represented by the pooled CNN features weighted by the objectness scores. We train two classifiers using features extracted from the signboard and region under signboard respectively. Finally, the probability values of two classifiers are treated as the image-level feature. The aggregation of image-level and text-level feature produces the final prediction. In our experiment, we observe that the accuracy of our proposed method is 91%, which is 10% higher than the state-of-art performance of 81% using GoogleNet [12,13].

Compared with traditional methods on scene classification, our proposed method utilizes the store–specific attributes (e.g., "edge boundary" and "repeatness") to find the discriminative regions

☆ This paper has been recommended for acceptance by Zicheng Liu.

* Corresponding author.
E-mail addresses: huangchao_uestc@aliyun.com (C. Huang), hlli@uestc.edu.cn (H. Li), weili.cv@gmail.com (W. Li), wqb.uestc@gmail.com (Q. Wu), lfxu@uestc.edu.cn (L. Xu).

http://dx.doi.org/10.1016/j.jvcir.2017.01.011
1047-3203/

for store classification. Meanwhile, we provide a new method to extract the text-level feature which gets rid of the requirement of the labeled character dataset. The combination of the image-level feature and text-level feature boosts the performance of store classification.

The rest of the paper is organized as follows. We review the previous scene classification methods in Section 2. Section 3 introduces the details of the proposed method. The experimental setup and results are provided in Section 4, followed by the conclusion.

2. Related work

Store recognition is a specific area of scene classification. In recent years, many works focused on scene classification have been conducted. These approaches can be generally divided into classification methods [14] based on low-level feature, methods using the semantic modeling to infer middle-level feature and methods based on neural network.

Methods using low-level feature: For traditional methods on scene classification [1], the low-level features such as Local Binary Patterns (LBP) [15–17], Histogram of Oriented Gradients (HOG) [18], SIFT [19] were firstly extracted from the dense patches of the images. Then, K-means method was employed to obtain the clusters of the low-level features. Each image was represented based on the histogram of visual words. Finally, the classifiers such as SVM [7] and random forest [8] were trained using BoW representation.

Methods based on semantic modeling: In recent years, several works have been proposed based on semantic modeling to infer the middle-level features. For instance, Wright et al. [20] proposed sparse coding to learn the compact feature representation. Blei et al. proposed Latent Dirichlet Allocation (LDA) which assigned a latent topic to each visual word. These latent topics captured the structure and semantic information. Huang et al. [10] constructed a graphical model which integrated the saliency map and appearance to extract the foreground [21,22]. Perronnin and Dance [5] employed GMM to model the relationship of low-level features and used fisher vector to represent the image.

Methods based on neural network: During the past few years, CNN feature based methods [23–25] have achieved the state-of-art performance on a large scale classification task [11]. These methods try to use architectures composed of non-linear transformations to extract middle-level features from images. Since there are tremendous parameters for CNN, we need a large amount of images to train the model. Fortunately, according to previous works [26,27], the pre-trained CNN models using the large-scale dataset can be transferred to the network for the specific task without enough training data. For scene classification, recent CNN feature based work [28] built a place dataset and used deep neural network to learn the CNN features. Compared with the performance of the pre-trained model using imageNet [11], it observed a significant improvement to distinguish different places. In [12], the GoogleNet [13] is employed to perform store recognition.

3. Our work

In this section, we introduce the proposed method consisting of three stages. The first is to obtain the text information (e.g. Text-Exemplar-Similarity) by calculating the similarity scores between the character candidates with the class-specific character exemplars. In the second stage, we extract the image-level feature (e.g. Hypotheses-Weighted-CNN) which is achieved by employing the selective search and CNN model. Finally, the image is predicted based on score aggregation. The framework is shown in Fig. 1. In what follows, the details of Text-Exemplar-Similarity, Hypotheses-Weighted-CNN and the score aggregation will be presented.

3.1. Text-Exemplar-Similarity

For store recognition, the text information plays an important role since some stores can be classified just based on the text in the store signboard. Traditional methods [29] generally focused on the text detection and character recognition using the labeled dataset. For instance, Coates et al. [30] proposed a method to automatically learn features in an unsupervised way. For text detection, these features were fed to train the classifier using the dataset with the "text" and "no text" labels. For character recognition, the labeled character dataset is required to train the classifier. Zhu et al. [31] incorporated the information derived from the text to improve the performance of classification, which is based on the information derived from the recognized characters. One drawback of these methods is that it's time consuming to collect the labeled dataset. In our work, we focus on a new direction which utilizes the text-level feature without recognizing the text. Our work is most related to that of Zamir et al. [32] which utilized the text-level feature to recognize business. It is noticed that this method still needs to generate a set of synthetic character patches. Compared with recent works, one advantage of our weakly supervised method is that we get rid of the labeled dataset for text detection or character

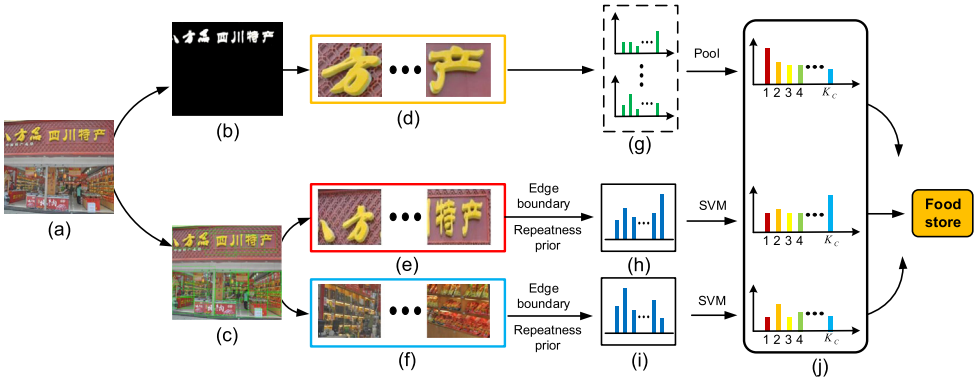

Fig. 1. The framework of the proposed method. (a) The input image. (b) The character candidates. (c) The object hypotheses based on selective search. (d) The image patches of the characters. (e) The object hypotheses in the signboard. (f) The object hypotheses under the signboard. (g) The highest similarity score with the character exemplars for each class. (h) The pooled CNN features weighted by edge boundary and repeatness prior of (e). (i) The pooled CNN features weighted by edge boundary and repeatness prior of (f). (j) The output scores of classifiers (K_C is the category number).

recognition. Our work consists of training the initial character detectors, selecting the discriminative characters and calculating the similarity scores between the candidate character with the discriminative character.

We first introduce the notations of variables and parameters used for extracting text-level feature, which are shown in Table 1. We use the bold letter to denote the vector.

The flowchart of learning the initial detectors is shown in Fig. 2, we first find the Maximally Stable Extremal Regions (MSER) [33] of the images. The output of MSER is shown in Fig. 2 (b). We can see that the candidate regions of characters can be found by using MSER method. Then, the noise regions are eliminated based on the aspect ratio, area size and stroke width [34,35]. For each candidate, we extract the HOG feature from the candidate region. Exemplar-SVM [36] is employed to train the detector of each candidate region. The trained detectors are shown in Fig. 2 (e). After the initial detectors are learned, we extract the HOG feature from the candidate regions. The similarity score between candidate region with detector can be calculated by W^TX+b, where X is the HOG feature, W is the weight of detector and b is the corresponding bias.

In the second stage, the discriminative character detectors are selected out from the initial detectors. We first collect the set of similarity scores for each detector. Then, we evaluate the j-th detector based on $F_{1,j}$ and $F_{2,j}$. The algorithm of choosing discriminative detectors is shown in Algorithm 1.

Algorithm 1. The algorithm of choosing discriminative detectors.

Require: The set of similarity scores for each initial detector $\{\mathbf{S}_1,\ldots,\mathbf{S}_{K_D}\}$ and the label of each image $\{c_1,\ldots,c_{K_I}\}$.
Ensure: The discriminative detector set for each category $\{\mathbf{o}_1,\ldots,\mathbf{o}_{K_C}\}$.
1: **repeat**
2: **for** $j=1$ to K_D **do**
3: We set the label of j-th detector l_j as the category of the image where the corresponding character is from.
4: We collect $\mathbf{P}_j$ and $\mathbf{N}_j$ by collecting the similarity scores from the positive and negative image set of l_j respectively.
5: We calculate $F_{1,j}=\sum_{i=1}^{K_I}\delta(P_{j,i}>T)$ and $F_{2,j}=\sum_{i=1}^{K_I}\delta(N_{j,i}>T)$, where $\delta(A)=1$ if A is true. Here, $P_{j,i}$ indicates the similarity score between the j-th initial detector with i-th image.
6: **if** $F_{1,j}>\lambda F_{2,j}$ **then**
7: The j-th detector is discriminative and we update the discriminative detector set of l_j by adding d_j to o_{l_j}.
8: **end if**
9: **end for**
10: **until** convergence

Given a set of image, we first collect the score set of discriminative detectors $\mathbf{o}$. Then, we find the highest similarity score for all classes $\boldsymbol{\beta}=\{\beta_1,\ldots\beta_k,\ldots,\beta_{K_C}\}$, where β_k is the highest similarity score for k-th class. The text-level feature $\mathbf{H}_t(I|k)$ can be written as follows:

$$\mathbf{H}_t(I|k)=\frac{1}{\sqrt{2\pi}\sigma}exp\left(\frac{\beta_k}{2\sigma^2}\right), \tag{1}$$

where we use gaussian function to introduce non-linearity to smooth $\boldsymbol{\beta}$. In our experiments, we set $\sigma=0.23$ experimentally. Finally, the classification is achieved by choosing the label with the maximum value in $\mathbf{H}_t(I)$. The parameters T and λ are optimized by cross validation.

Table 1
Notations of variables and parameters for extracting text-level feature.

Parameters/variables	Notations
$c_i\in\{1,\ldots,K_C\}$ where K_C is the number of category.	The category of the i-th image
$I_j\in\{1,\ldots,K_C\}$	The category of the j-th initial trained detector
$I_i\in\{1,\ldots,K_I\}$ where K_I is the number of image.	The i-th image
$d_j\in\{1,\ldots,K_D\}$ where K_D is the number of the initial detectors.	The j-th initial detector
$\boldsymbol{S_j}\{S_{j,1},\ldots,S_{j,k_I}\}$	The similarity score set for the j-th initial detector
$\boldsymbol{P_j}\{P_{j,1},\ldots,P_{j,k_I}\}$ and $\boldsymbol{N_j}\{N_{j,1},\ldots,N_{j,k_I}\}$	The collected similarity scores from the positive and negative images of l_j
$F_{1,j}$ and $F_{2,j}$	The number of similarity scores higher than a threshold in $\boldsymbol{P_j}$ and $\boldsymbol{N_j}$ respectively
T and λ	The parameters of threshold
$\boldsymbol{o_k}$	The discriminative detector set of the k-th category
$\boldsymbol{\beta_k}$	The similarity score set with $\boldsymbol{o}_k$

3.2. Hypotheses-Weighted-CNN

In our work, the Hypotheses-Weighted-CNN feature consists of five steps: object hypotheses generation, CNN feature extraction, object score calculation, signboard location and feature pooling. The flowchart is shown in Fig. 3.

The first step is to generate the regions for probable objects. In our work, the region proposal method named Selective Search [37] is employed since the region hypotheses generated by this method have strong objectness. According to [38], the Intersection over Union (IoU) value with the ground-truth object region is higher than most of the region proposal methods with the comparable region number. This method proposed a hierarchical grouping method to generate possible object locations.

After obtaining the object hypotheses, we extract the features from each region. Here, the framework of Convolution Neural Network (CNN) is utilized to represent each candidate region. Compared with the hand-crafted features, CNN feature is a learnt image feature with deep network structure, which has shown state-of-art performance in many computer vision tasks such as image classification and object detection. In our work, each region is represented by Place-CNN feature [28], which trains the CNN model on a large place dataset. The network architecture of Places-CNN uses five convolutional layers followed by three fully connected layers. Since the feature of each region is based on the output of the 6-th layer which is fully connected and has 4096 neurons in this layer, so the feature dimension is 4096.

In the third step, we use two priors to calculate the objectness score, namely "edge boundary" and "repeatness". The first prior is based on the assumption that a good proposal which encloses the object should have little edge in the boundary of the bounding box. The motivation is that the edge in the box boundary will be strong if the object proposal just covers part of object. We call it "edge boundary". Given an input image, we use the structure forest to detect the edge map [39–41]. For each object box, the edge values in the boundary of the box are denoted as E. The equation to compute the score of the first prior can be written as follows:

$$\eta=\exp\left(-\rho\frac{\Sigma E}{h+w}\right), \tag{2}$$

where h and w are the height and width of the box. In our experiment, we set $\rho=0.2$. Some examples chosen based on the *edge boundary prior* are shown in Fig. 4. It can be seen that the boxes enclose the object when the score of edge boundary prior is high.

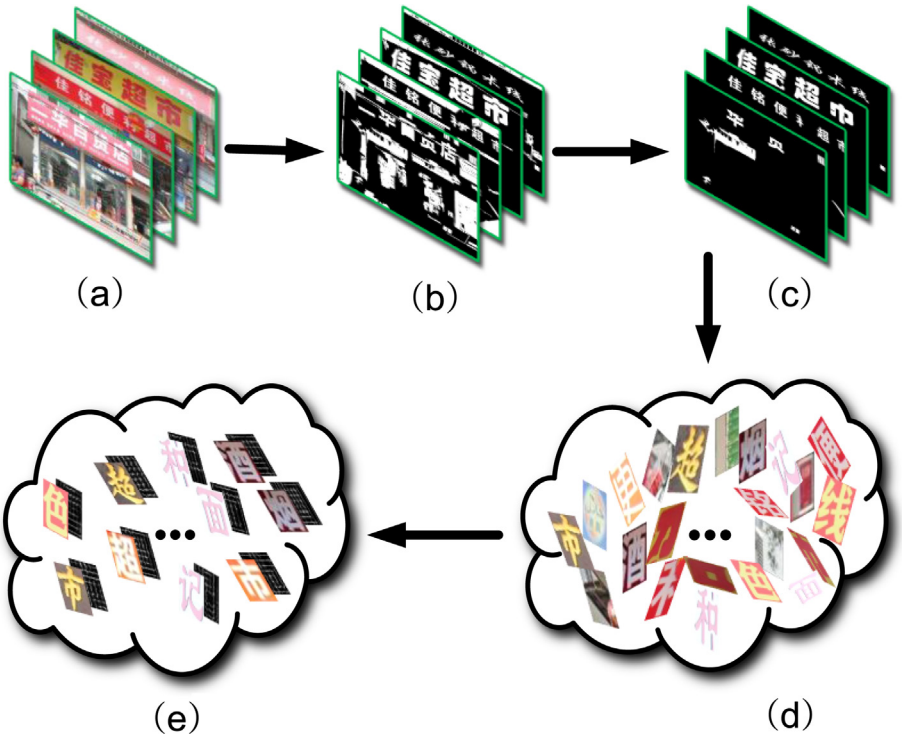

Fig. 2. The flowchart to train the character detectors. (a) The training images. (b) The results using MSER method. (c) The character candidates. (d) The image patches of the character candidates. (e) The initially trained detectors using HOG feature.

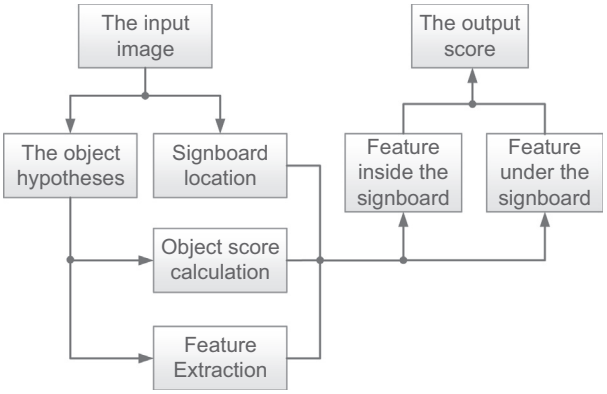

Fig. 3. The flowchart to extract the image-level feature.

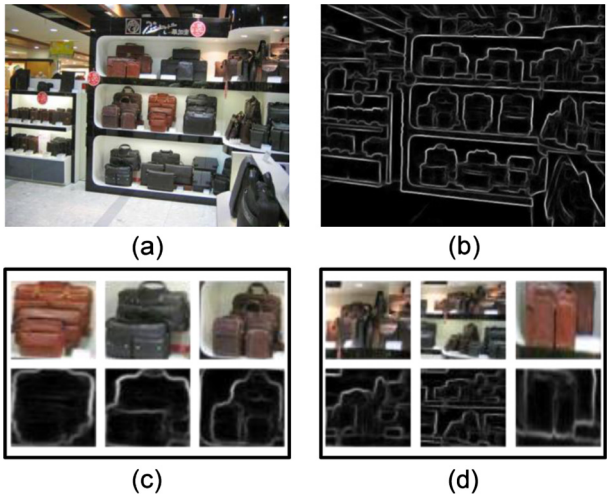

Fig. 4. (a) The input image. (b) The edge map detected by the structure forest. (c) The object hypotheses with high edge boundary prior. (d) The object hypotheses with low edge boundary prior.

The second prior is from the observation that the commodities in a store generally share similar shape. In other words, the object hypotheses which share appearance with others are more likely to be the commodities. This prior is named by "repeatness". To extract the shape feature, we first collect the dense patches of the edge map inside the candidate. Then, each edge patch is quantized using the Bag-of-Words (BoW) framework. Finally, the object candidate is represented by the histogram of the visual words, which is regularized by $L1$-norm. We use the histogram intersection kernel to measure the similarity of the object hypotheses. The final score computation is achieved by averaging the edge boundary scores of K_N nearest neighbors based on "repeatness". We set $K_N = 5$ experimentally in our experiment.

In the fourth step, we locate the position of the signboard based on MSER [33] and noise elimination. This procedure uses the MSER obtained in Section 3.1. Each region is dilated via the morphological operation with line structuring element. The connected regions whose area size is larger than a threshold are chosen as the box candidate. The noise elimination of the candidates is achieved based on the location and aspect ratio.

Given the signboard location, we compute the overlap ratio between the object hypotheses with the signboard. Here, the overlap degree is measured by the ratio between the intersecting area and the area of the object hypotheses. According to the overlap ratio with the signboard, we collect the object bounding boxes inside the signboard. The representation $\mathbf{Q}$ (inside the signboard) is achieved by pooling the CNN features f of object hypotheses based on the objectness score θ considering the edge boundary score and repeatness score. The function to obtain $\mathbf{Q}_j$ can be written as:

$$\mathbf{Q}_j = max(\theta_1 f_j^1, \ldots, \theta_{K_O} f_j^{K_O}),\ for\ j = \{1, \ldots, 4096\}, \quad (3)$$

where K_O is the number of the object hypotheses inside the signboard, f_j^i means the j-th feature value of ith object candidate. We use similar method to represent the region under the signboard, which is depicted as $\hat{\mathbf{Q}}$. If the signboard is not detected in the image, we set $\mathbf{Q} = \mathbf{0}$ and use all proposals to generate $\hat{\mathbf{Q}}$.

The linear SVM is used to train the classifiers based on the pooled feature $\mathbf{Q}$ and $\hat{\mathbf{Q}}$ respectively. The outputs of the classifiers are treated as the intermediate features, which are depicted as $\mathbf{H}_i(I)$ and $\mathbf{H}_u(I)$. Here, the dimension of $\mathbf{H}_i(I)$ and $\mathbf{H}_u(I)$ is K_C.

3.3. Combined classifier

The discriminative power of text and image-level features ($\mathbf{H}_t, \mathbf{H}_i$ and $\mathbf{H}_u$) varies for different categories. According to [9], the improved performance will be achieved by combining different aspects of feature. In our work, we combine the middle features $\mathbf{H}_t, \mathbf{H}_i$ and $\mathbf{H}_u$ to perform store classification. Given a test image, the energy for ith class is:

$$\log \mathbf{H}(I|k) = \alpha_t \log \mathbf{H}_t(I|k) + \alpha_i \log \mathbf{H}_i(I|k) + \alpha_u \log \mathbf{H}_u(I|k), \quad (4)$$

where $\boldsymbol{\alpha} = \{\alpha_t, \alpha_i, \alpha_u\}$ is weights of $\{\mathbf{H}_t, \mathbf{H}_i, \mathbf{H}_u\}$. The label of the test image can be written as follows:

$$k^* = arg\ \max_k \{\mathbf{H}(I|k)\}. \quad (5)$$

We employ stacked learning [42–44] to generate the validation set. After the generation of the validation set, the weight optimization is achieved by maximizing the classification accuracy on the validation set. Note that there are only two independent parameters in the weight $\boldsymbol{\alpha}$. We set $\alpha_u = 1$ and employ cross validation to find the rest optimized parameters.

4. Experiment

In this section, we first introduce a new store dataset. We verify our proposed method on our dataset and Street28 dataset [45]. The experimental setup and results are reported.

4.1. Dataset

We introduce a dataset consisting of 9 store categories called by "Store9". The dataset is collected from camera photos and internet images. There are total 1090 images: 107 bags store, 115 bookstore, 128 clothing store, 103 drugstore, 113 electrical-store, 180 furniture store, 143 food store, 101 shoe store and 100 super market. As shown in Fig. 5, this store dataset is very challenging. Here, we list several difficulties of this dataset.

- There is a large number of variation of illumination, view, scale and occlusion.
- The background is extremely cluttered and complex.
- The area of store is small in some images.
- The inter-class variation is sometimes smaller than the intra-class variation.

We also provide the groundtruth labels for all images of the dataset.

4.2. Experiment result

Store9 We aim to classify these 9 stores using the combined image and text-level feature. There are nearly 100 images for each category. We use about half (50) images for training and the rest

Fig. 5. The images of different store categories. We show three images from the same category in each row.

images for testing. We compare the performance of the proposed method with some baseline methods. For BoW method [2], the low-level features (color SIFT) are extracted from multi-scale {4, 8, 16, 24} patches on a grid with stepsize 8 pixels. Then, K-means clustering method [46] is utilized to find 512 clusters. Each descriptor is quantized to the nearest cluster. The histogram of the clusters is used to represent the image. For FV [5], we reduce the dimension to 64 using principal component analysis (PCA) [47] after the generation of the low-level features. We train a Gaussian Mixture Model (GMM) to approximate the distribution of the low level features. Then, the Fisher Vector is employed to represent the image. For CNN-based methods, we employ the CNN model [48] pre-trained on imageNet [11], the Place-CNN model [28], and GoogleNet [12]. All networks are fine-tuned by our dataset. The learning rate is set as 0.001.

The classification accuracy of different methods are shown in Table 2. It indicates that the Place-CNN feature is more suitable for store classification. The accuracy is nearly 5% higher than that of imageNet-CNN feature. The worst-performing method is BoW due to the high quantization error. It also shows that the auto-

Table 2
Classification accuracy in Store9 dataset.

Model	Accuracy (%)
SPM [2]	44.22
Fisher Vector [5]	58.91
ImageNet-CNN [49]	64.84
Place-CNN [28]	69.22
GoogleNet [12]	80.9
Proposed method	90.98

learnt feature is more discriminative than the hand-crafted features. Using the feature extracted by GoogleNet obtains 80.9%. The best performance is achieved by our method, which is nearly 10% higher than the state-of-art result using GoogleNet. Given a test image, it takes about 1.2 seconds to obtain the predicted label. we use Matlab 2015b and one TitanX to evaluate our code. The deep features extracted from proposals take about 1 MB for each image. Since we extract the deep features from proposals of the image, the time complexity and the space are N_p times larger than traditional CNN method, where N_p is the number of proposals. In future, we can make these proposals share the output of convolutional layer, which can save time and space.

Street28 This dataset consists 28 classes collected from the ImageNet subclasses of place and building. There are 24255 images in this dataset. We split the dataset into the training and test set according to the experimental setup of [45]. The text-level and image-level features are integrated to perform image classification. For the text-level feature, we select a set of discriminative detectors from the initial text detectors. Given a test image, the text-level features are extracted based on the similarity scores with the discriminative detectors. For the image-level feature, since there are only a few images containing a signboard in this dataset, it is unable to extract the deep features (output of the 6th fully connected layer of the fine-tuned alex-net [26]) from proposals in and under the signboard. Instead, we first detect the text in each image. Then, we generate a set of proposals by selective search. Each proposal is weighted based two introduced priors. These proposals are divided into two kinds according to whether the proposals contain text. We obtain the representation of two kinds by pooling the weighted deep features. The pooled features are treated as the image-level feature. Finally, we train linear SVM based on the image-level and text-level features. The result is shown in Table 3. We compared our method with the work of Karaoglu [45] and fine-tuned alex-net [26]. It shows that our method obtains the best performance compared with recent methods. Using the text-level feature can further improve the performance by 3.3%.

4.3. Discussion

In this section, we investigate the influence of each term, followed by the report of the robustness to the noise. Our experiment is evaluated in store9, since each image of this dataset consists the signboard and the text information.

To investigate the efficiency of feature pooling, we first generate a set of object hypotheses. Then, the CNN features are employed to represent each candidate. We use the max-pooled feature to represent the image. The performance is compared with the accuracy using Place-CNN feature of the image. The result is shown in Table 4. It shows that using the pooled feature significantly improve the performance by nearly 15%. This is probably because the hypotheses enclosing the object has better discriminant power than the whole image.

We also investigate how the objectness score influences the performance of classification. The objectness score of each object candidate is calculated based on the edge boundary and repeatness. For each candidate, we find the nearest neighbors based on the shape similarity. Then, the pooling weight is calculated based on the averaged value of edge boundary score. The accuracy is 85.60% which is 1.4% higher than the pooled feature. This method is compared with using the saliency map to weight each proposal, we generate the saliency map based on the method of cheng et al. [50]. Then, each proposal is weighted based on the mean saliency value. From Table 4, it can be seen that our method is 1.1% higher than saliency based method. The reason is perhaps that our method utilizes the attribute of store and can give higher weight to proposal containing object. The results indicates the efficiency of objectness score, which eliminates most of the effects of cluttered background.

Then, we investigate the effect of signboard location for store classification. In our experiment, the object candidates inside and under the signboard are represented by the pooled CNN feature weighted by the objectness score. We train two classifiers for these features. The outputs are aggregated to classify the image. The accuracy is 87.03%, which indicates that the features inside and under the signboard are complementary to achieve better performance.

The effect of the text information is also investigated. By combining Text-Exemplar-Similarity with Hypotheses-Weighted-CNN feature, the best performance is achieved, which demonstrates the importance of text information for store classification. The reason is that some stores which share similar image-level features can be distinguished from each other by text information.

To show the efficiency of each feature, We report the performance of using only one feature. The results of using only pooled feature under signboard, inside signboard and text-level feature are shown in Table 5. We find that using the image-level feature under the signboard obtains the best performance. The reason is perhaps that the object inside the store can provide discriminative information to distinguish different categories. According to the result of "CNN+obj" shown in Table 4, we can see that combining the feature under and inside the signboard can further increase the performance by about 2%. The reason for pool performance obtained for the text-level feature is that there exists a large variation of text information inside the signboard for many classes such as shoes and cloth stores. While the combination of image-level and text-level feature increases the performance remarkably, which indicates that the textual cue can bring new discriminative information for store classification.

Table 3
Classification accuracy in Street28 dataset.

Model	Accuracy (%)
Karaoglu [45]	39.0
Alex-net [26]	47.8
Proposed method (image-level)	52.0
Proposed method (all)	55.3

Table 4
Classification accuracy ("CNN+maxpool" denotes that the max-pooled CNN features are employed for image classification. "CNN+saliency" means that the weight of each proposal is based on the saliency map. "CNN+obj" means the method that we use the objectness score to help the feature pooling of CNN features extracted from object hypotheses. "CNN+obj+signboard" means that the pooled feature inside and under the signboard are integrated to classify the image. "CNN+obj+singboard+text" denotes the method which combines Text-Exemplar-Similarity with Hypotheses-Weighted-CNN feature to represent the image.)

Model	Accuracy (%)
Place-CNN	69.22
CNN+maxpool	84.18
CNN+saliency	84.50
CNN+obj	85.60
CNN+obj+signbord	87.03
CNN+obj+signbord+text	90.98

Table 5
Classification accuracy using various features.

Model	Accuracy (%)
Deep feature under signboard	85.22
Deep feature inside signboard	61.33
Text-level feature	49.67
All	90.98

We also show the classification performance of our method when we add some Gaussian random noise distributed as $p \sim N(0; 20)$ to each image. The accuracy by using the image-level feature is 86.1%. It obtains 90.1% by using all features. From these results, we observe that the performance of our method drops slightly when the noise is added to the image. The main reason is perhaps that using pooled deep feature is robust to the noise to some extent.

5. Conclusions

In this paper, we focus on the store classification, which is a challenging task due to the cluttered background. Therefore, we introduce a simple and efficient approach to eliminating the noise of the background, which is based on the object candidate and the objectness score. The CNN feature extracted from the candidate with a high objectness score plays a more important role for feature pooling. In our experiment, we observed a significant improvement over the method using the CNN feature of the whole image. The pooled features inside and under the signboard are integrated for store classification. The result shows that better performance can be achieved by combing two features. Additionally, we investigate the efficiency of text-level features. For each category, we learn a set of discriminant character detectors using exemplar-SVM. Given a test image, the similarity scores with these detectors are treated as the middle features. Despite the simplicity of our proposed method, it achieves the best performance compared with the state-of-art method, which uses GoogleNet to extract the global feature. In our future work, we are interested in exploiting the relationship of the characters, which may further improve the discriminative power of text-level feature. In addition, foreground segmentation might help reduce the noise of cluttered background.

Acknowledgment

This work was supported in part by National Natural Science Foundation of China (Nos. 61525102, 61601102, 61502084).

References

[1] A. Oliva, A. Torralba, Modeling the shape of the scene: a holistic representation of the spatial envelope, Int. J. Comput. Vis. 42 (3) (2001) 145–175.
[2] S. Lazebnik, C. Schmid, J. Ponce, Beyond bags of features: spatial pyramid matching for recognizing natural scene categories, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), vol. 2, New York, NY, USA, 2006, pp. 2169–2178.
[3] J. Shi, X. Li, Y. Dong, How to represent scenes for classification?, in: 2015 IEEE China Summit and International Conference on Signal and Information Processing (ChinaSIP), IEEE, 2015, pp 191–195.
[4] L.-J. Li, L. Fei-Fei, What, where and who? classifying events by scene and object recognition, in: IEEE International Conference on Computer Vision (ICCV), Rio de Janeiro, Brazil, 2007, pp. 1–8.
[5] F. Perronnin, C. Dance, Fisher kernels on visual vocabularies for image categorization, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Minneapolis, Minnesota, USA, 2007, pp. 1–8.
[6] X. Zhou, K. Yu, T. Zhang, T.S. Huang, Image classification using super-vector coding of local image descriptors, in: European Conference on Computer Vision (ECCV), Springer, 2010, pp. 141–154.
[7] M.A. Hearst, S. Dumais, E. Osman, J. Platt, B. Scholkopf, Support vector machines, Intell. Syst. Appl. 13 (4) (1998) 18–28.
[8] L. Breiman, Random forests, Machine Learn. 45 (1) (2001) 5–32.
[9] M.-E. Nilsback, A. Zisserman, A visual vocabulary for flower classification, IEEE Conference on Computer Vision and Pattern Recognition (CVPR), vol. 2, IEEE, 2006, pp. 1447–1454.
[10] C. Huang, F. Meng, W. Luo, S. Zhu, Bird breed classification and annotation using saliency based graphical model, J. Visual Commun. Image Represent. 25 (6) (2014) 1299–1307.
[11] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, L. Fei-Fei, Imagenet: a large-scale hierarchical image database, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, 2009, pp. 248–255.
[12] Y. Movshovitz-Attias, Q. Yu, M.C. Stumpe, V. Shet, S. Arnoud, L. Yatziv, Ontological supervision for fine grained classification of street view storefronts, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015, pp. 1693–1702.
[13] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, A. Rabinovich, Going deeper with convolutions (2015) 1–9.
[14] Y. Dong, D. Tao, X. Li, Nonnegative multiresolution representation-based texture image classification, ACM Trans. Intell. Syst. Technol. (TIST) 7 (1) (2015) 4.
[15] T. Ojala, M. Pietikainen, T. Maenpaa, Multiresolution gray-scale and rotation invariant texture classification with local binary patterns, IEEE Trans. Pattern Anal. Machine Intell. 24 (7) (2002) 971–987.
[16] T. Song, H. Li, Local polar dct features for image description, IEEE Signal Process. Lett. 20 (1) (2013) 59–62.
[17] T. Song, H. Li, Wavelbp based hierarchical features for image classification, Pattern Recogn. Lett. 34 (12) (2013) 1323–1328.
[18] N. Dalal, B. Triggs, Histograms of oriented gradients for human detection, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), vol. 1, San Diego, CA, USA, 2005, pp. 886–893.
[19] D. Lowe, Object recognition from local scale-invariant features, in: IEEE International Conference on Computer Vision (ICCV), vol. 2, Kerkyra, Corfu, Greece, 1999, pp. 1150–1157. doi:http://dx.doi.org/10.1109/ICCV.1999.790410.
[20] J. Wright, Y. Ma, J. Mairal, G. Sapiro, T.S. Huang, S. Yan, Sparse representation for computer vision and pattern recognition, Proc. IEEE 98 (6) (2010) 1031–1044.
[21] H. Li, K.N. Ngan, A co-saliency model of image pairs, IEEE Trans. Image Process. 20 (12) (2011) 3365–3375.
[22] H. Li, F. Meng, K.N. Ngan, Co-salient object detection from multiple images, IEEE Trans. Multimedia 15 (8) (2013) 1896–1909.
[23] N. Zhang, J. Donahue, R. Girshick, T. Darrell, Part-based r-cnns for fine-grained category detection, in: European Conference on Computer Vision (ECCV), Springer, 2014, pp. 834–849.
[24] R. Girshick, J. Donahue, T. Darrell, J. Malik, Rich feature hierarchies for accurate object detection and semantic segmentation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2014, pp. 580–587.
[25] X. Jiang, Y. Pang, X. Li, J. Pan, Speed up deep neural network based pedestrian detection by sharing features across multi-scale models, Neurocomputing 185 (2016) 163–170.
[26] A. Krizhevsky, I. Sutskever, G.E. Hinton, Imagenet classification with deep convolutional neural networks, in: Advances in Neural Information Processing Systems, 2012, pp. 1097–1105.
[27] Y. Wei, W. Xia, J. Huang, B. Ni, J. Dong, Y. Zhao, S. Yan, Cnn: single-label to multi-label, arXiv:1406.5726.
[28] B. Zhou, A. Lapedriza, J. Xiao, A. Torralba, A. Oliva, Learning deep features for scene recognition using places database, in: Advances in Neural Information Processing Systems, 2014, pp. 487–495.
[29] C. Shi, C. Wang, B. Xiao, Y. Zhang, S. Gao, Z. Zhang, Scene text recognition using part-based tree-structured character detection, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2013, pp. 2961–2968.
[30] A. Coates, B. Carpenter, C. Case, S. Satheesh, B. Suresh, T. Wang, D.J. Wu, A.Y. Ng, Text detection and character recognition in scene images with unsupervised feature learning, in: 2011 International Conference on Document Analysis and Recognition (ICDAR), IEEE, 2011, pp. 440–445.
[31] Q. Zhu, M.-C. Yeh, K.-T. Cheng, Multimodal fusion using learned text concepts for image categorization, in: Proceedings of the 14th ACM International Conference on Multimedia, ACM, 2006, pp. 211–220.
[32] A.R. Zamir, A. Dehghan, M. Shah, Visual business recognition: a multimodal approach., in: ACM Multimedia, Citeseer, 2013, pp. 665–668.
[33] J. Matas, O. Chum, M. Urban, T. Pajdla, Robust wide-baseline stereo from maximally stable extremal regions, Image Vis. Comput. 22 (10) (2004) 761–767.
[34] H. Chen, S.S. Tsai, G. Schroth, D.M. Chen, R. Grzeszczuk, B. Girod, Robust text detection in natural images with edge-enhanced maximally stable extremal regions, in: IEEE International Conference on Image Processing (ICIP), IEEE, 2011, pp. 2609–2612.
[35] B. Epshtein, E. Ofek, Y. Wexler, Detecting text in natural scenes with stroke width transform, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, 2010, pp. 2963–2970.
[36] T. Malisiewicz, A. Gupta, A.A. Efros, Ensemble of exemplar-svms for object detection and beyond, in: IEEE International Conference on Computer Vision (ICCV), IEEE, 2011, pp. 89–96.
[37] K.E. Van de Sande, J.R. Uijlings, T. Gevers, A.W. Smeulders, Segmentation as selective search for object recognition, in: IEEE International Conference on Computer Vision (ICCV), IEEE, 2011, pp. 1879–1886.
[38] C. Wang, W. Ren, K. Huang, T. Tan, Weakly supervised object localization with latent category learning, in: European Conference on Computer Vision (ECCV), Springer, 2014, pp. 431–445.
[39] P. Dollár, C. Zitnick, Structured forests for fast edge detection, in: IEEE International Conference on Computer Vision (ICCV), 2013, pp. 1841–1848.
[40] P. Dollár, C.L. Zitnick, Fast edge detection using structured forests, IEEE Trans. Pattern Anal. Machine Intell. 37 (8) (2015) 1558–1570.
[41] C.L. Zitnick, P. Dollár, Edge boxes: locating object proposals from edges, in: European Conference on Computer Vision (ECCV), Springer, 2014, pp. 391–405.

[42] D.H. Wolpert, Stacked generalization, Neural Netw. 5 (2) (1992) 241–259.
[43] W.W. Cohen, Stacked sequential learning, Tech. rep., DTIC Document, 2005.
[44] R. Varun, M. Daniel, H. Martial, B.J. Andrew, S. Yaser, Pose machines: articulated pose estimation via inference machines, in: European Conference on Computer Vision (ECCV), 2011, pp. 1832–1839.
[45] S. Karaoglu, J.C. van Gemert, T. Gevers, Context: text detection using background connectivity for fine-grained object classification, in: Proceedings of the 21st ACM International Conference on Multimedia, ACM, 2013, pp. 757–760.
[46] A. Vedaldi, B. Fulkerson, VLFeat: an open and portable library of computer vision algorithms, 2008. <http://www.vlfeat.org/>.
[47] Y. Ke, R. Sukthankar, Pca-sift: a more distinctive representation for local image descriptors, IEEE Conference on Computer Vision and Pattern Recognition (CVPR), vol. 2, IEEE, 2004, pp. II–506.
[48] Y. Jia, Caffe: an open source convolutional architecture for fast feature embedding, in: http://caffe.berkeleyvision.org/, 2013.
[49] J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, T. Darrell, Decaf: a deep convolutional activation feature for generic visual recognition, arXiv:1310.1531.
[50] M.-M. Cheng, N.J. Mitra, X. Huang, P.H. Torr, S.-M. Hu, Global contrast based salient region detection, IEEE Trans. Pattern Anal. Machine Intell. 37 (3) (2015) 569–582.