A Two-Stage Approach for Bag Detection in Pedestrian Images

Yuning Du[1], Haizhou Ai[1] and Shihong Lao[2]

[1]Computer Science and Technology Department, Tsinghua University
[2]OMRON Social Solutions Co., LTD

Abstract. Bag detection in pedestrian images is a very practical visual surveillance problem. It is challenging because bag appearance may vary greatly. In this paper, we propose a novel two-stage approach for bag detection in pedestrian images. Firstly, we utilize two stripe vocabulary forests to check whether a pedestrian is with a bag. Secondly, we locate the bag location by ranking the generated bottom-up region proposals. The ranker is learned with a convolutional neural network (CNN). Experiments are performed on a subset of CUHK person re-identification dataset that show the effectiveness of our approach for bag detection in pedestrian images. Although developed for a specific problem, our approach could be applied to detect other carrying objects in pedestrian images.

1 Introduction

In visual surveillance, people are interested in automatically searching persons from a huge amount of video data [1–11]. Because bag is a very common target appeared in surveillance video from public areas such as streets, subways, tourist attractions, airports and supermarkets, mining bag information is conducive to criminals monitoring, lost person search, video index and criminal investigation, and so on.

Bag detection plays an important role in bag information extraction. Firstly, it can greatly reduce the number of candidates for person searching when we only concern whether a person is with a bag. Secondly, it can also be used for the abnormal event detection, such as losing bag and stealing bag. Moreover, it provides prior knowledge for high level bag information extraction, such as bag color and type recognition, and so on. For convenience, in this paper, pedestrian images with bag will be called as bag images and those without bag as non-bag images. Some bag images and non-bag ones are shown in Fig.1. When the bag area is too small, as seen in Fig.1(c), although those pedestrians are with bags, they contain less bag information and it is hard to utilize. Besides, a person usually carries with one bag. Therefore, we mainly focus on the bag images where there is only one bag and more than 50% the bag area is visible. Because bag appearance changes due to variations in bag type, illumination, pedestrian pose and background clutter, bag detection is a challenging problem.

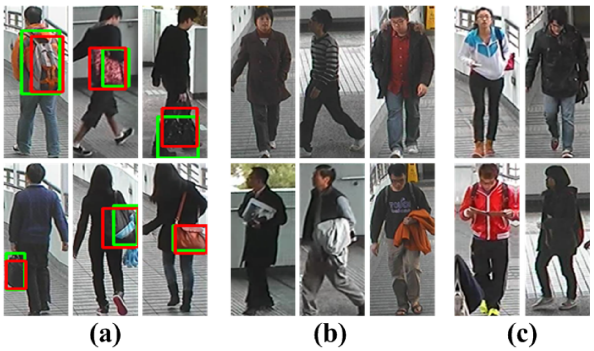


Fig. 1. (a) Some bag images. (b) Some non-bag images. (c) Some bag images where bag area is too small. The red boxes are ground truth of bag locations and the green ones are the detection results of our approach. All images in this paper are from CUHK person re-identification dataset [4].

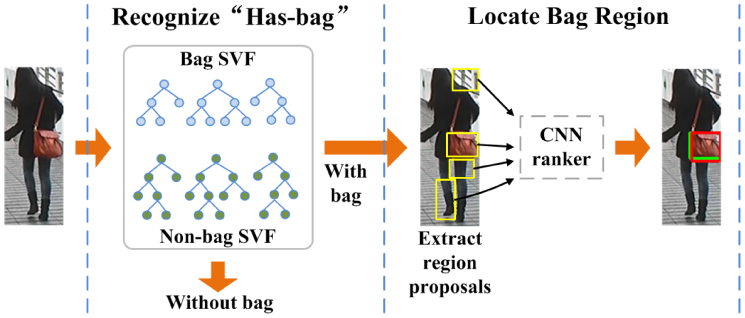


Fig. 2. Illustration of the framework of our approach.

Although there are some previous works [12–15, 29–31] on bag detection in short video sequences, to the best of our knowledge, no pervious work studies on this problem in a pedestrian image. It is also a valuable problem and more challenging than bag detection in short video sequences due to the less available information. Firstly, to extract pedestrian bag information (such as bag color, bag type) to retrieve a certain person from his tracklets, one effective way is by extracting bag information from some key pedestrian images of the tracklets. Then extracting bag information in the images is important. Secondly, bag detection in images is an important ingredient for tracking-by-detection approaches. Thirdly, bag detection in images is a specific problem, it provides insights on feature design and learning approaches.

In this paper, we propose a novel two-stage approach for bag detection in pedestrian images, as seen in Fig.2. Firstly, we check whether a pedestrian is with a bag, i.e. recognize pedestrian attribute "has-bag". Secondly, we locate

the bag region when there is a bag around the pedestrian, i.e. locate bag region. To recognize "has-bag", we utilize the particular structure of pedestrian and construct two different stripe vocabulary forests (SVFs) from bag images and non-bag ones. Stripes are rectangle regions whose width are the same as that of the pedestrian image and they are widely used in person re-identification [5, 8, 9, 11]. We estimate the likelihood of each stripe of the test image containing bag regions by SVFs. Then, combine those likelihoods of the above stripes to recognize "has-bag". When a pedestrian is with a bag, a small number of bottom-up region proposals will be extracted by selective search algorithm [16]. Those region proposals give high quality locations of bag. Then, we rank those region proposals with a convolutional neural network and regard the top 1 region proposal as the location of bag. Our main contribution is a novel two-stage approach for bag detection in pedestrian images and we show the effectiveness of our approach on a subset of CUHK person re-identification dataset. Although our approach is developed for bag detection in pedestrian images, it can be also applied to detect other carrying objects.

The rest of the paper is organized as follows. In Sec.2, we present the related work of our approach. In Sec.3, we show how to recognize "has-bag" with the stripe vocabulary forests. In Sec.4, we show how to locate the bag region with selective search and CNN. In Sec.5, we give and discuss experimental results. In Sec.6, we conclude this paper.

2 Related work

There are some previous works [10, 11, 17–19] on other pedestrian attributes recognition in a person image for "is-male", "has-hat", "has-shorts", "has-vnecks", "pedestrian orientation", etc. The approaches for pedestrian attributes recognition can be broadly categorized in two directions. In one direction, some works train discriminative models using Support Vector Machine (SVM) [11], Adaboost [19] and Random Forest [18] to recognize pedestrian attributes with a feature vector extracted from full body images. However, they neglect that bag regions exist in bag images. Those bag regions have stronger capacity for distinguishing bag images and non-bag ones than other regions. In another direction, some works utilize body segmentation and pose estimation [10, 17] to recognize person attributes. But it is hard to take advantage of them to recognize "has-bag". There are two main reasons. Firstly, "has-bag" is different from other pedestrian attributes such as "has-hat". Human usually wears a hat on the head, but the location of the bag in images varies dramatically and bag may appear anywhere in bag images, as seen in Fig.1(a). Secondly, bag may not appear in human body regions.

A sliding window approach for object detection is common. However, because there are large number of candidate windows to be distinguished, the sliding window approach is hard to use for bag detection. Recently, many works [16, 20, 21] attempt to generate a small set of high quality windows. Since bag regions in pedestrian images are always compact, it is easier to recall the bag regions

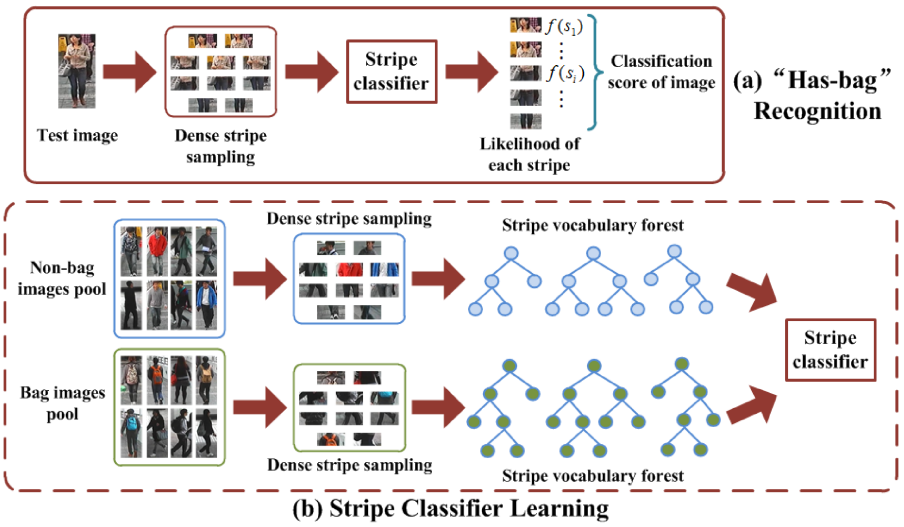


Fig. 3. (a) The framework of "has-bag" recognition in our approach. (b) The flowchart of the stripe classifier learning with two stripe vocabulary forests.

by those bottom-up region proposal generation approaches. Nevertheless, this is merely a preprocessing step and lots of non-bag regions need to be filtered out. Krizhevsky et al.[22] show that deep convolutional neural network achieved better results on the 2012 ImageNet Large Scale Visual Recognition Challenge (ILSVRC) using purely supervised learning. Inspired from their work, in this paper we learn a ranker with a convolutional neural network to rank the region proposals and locate the bag region.

3 Recognize "has-bag" with stripe vocabulary forests

Fig.3 shows the framework of "has-bag" recognition in our approach with two stripe vocabulary forests. Given a test image p, at first, we densely sample stripes from the image, denoted as $S^p = \{s_1^p, s_2^p, \cdots, s_{n_s}^p\}$. Then we estimate the likelihood of each stripe in S^p containing bag regions with a stripe classifier and combine those likelihoods to recognize "has-bag". To learn the stripe classifier, we densely sample stripes from non-bag images to form the negative stripe set, denoted as Θ^{N}. Similarly the positive stripe set is from bag images, denoted as Θ^{P}. Then, we build two stripe vocabulary forests from Θ^{P} and Θ^{N} separately and learn the stripe classifier with the above two stripe vocabulary forests.

3.1 Stripe vocabulary forest construction

Given a stripe set Θ, we apply the hierarchical k-means algorithm presented in [24, 25] to build a stripe vocabulary tree T from Θ and estimate the similarity

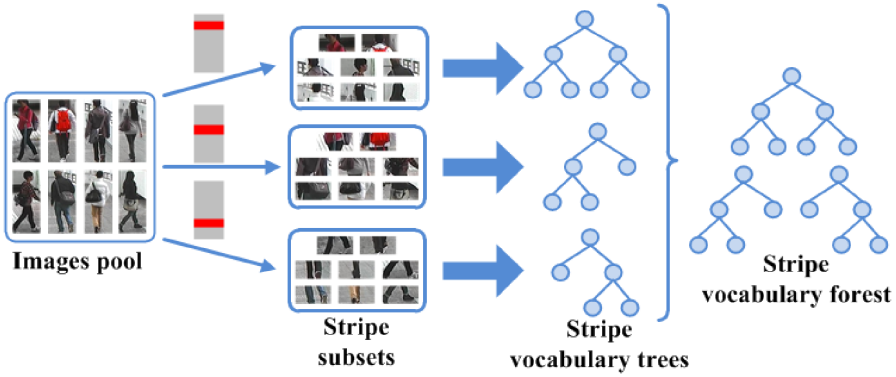


Fig. 4. Illustration of the stripe vocabulary forest construction.

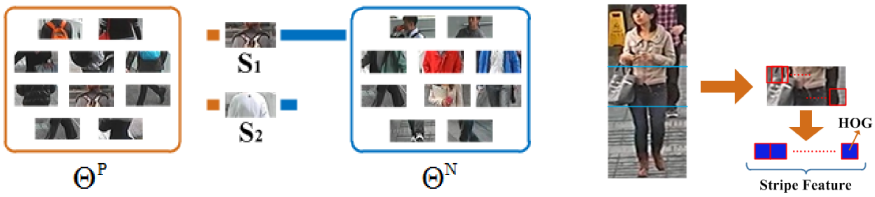
$sim(s, \Theta)$ between one stripe s and the stripe set Θ. The hierarchical k-means algorithm is scalable and to achieve good performance in other image retrieval problems. Send the stripes in Θ to the stripe vocabulary tree, the stripes reach the same leaf node will form a stripe set. Then, each leaf node will associate with a stripe set, denoted as $\{\tilde{S}_1^T, \tilde{S}_2^T, \cdots, \tilde{S}_m^T\}$, where m is the number of the leaf node in the stripe vocabulary tree T.

Estimating $sim(s, \Theta)$ only with the stripe vocabulary tree constructed from all stripes in Θ ignores the particular structure of pedestrian. Intuitively, although the pose of pedestrian may vary dramatically, vertical misalignment of two pedestrian images is slight. For example, the vertical range of head, upper body and lower body are similar between two different pedestrian images. Thus we build a stripe vocabulary forest instead of above stripe vocabulary tree by utilizing the particular structure of pedestrian, as seen in Fig.4. When we densely sample stripes from an image, assume h_i is the center position of the ith stripe in vertical. At first, we generate stripe subsets $\{\Theta_1, \Theta_2, ..., \Theta_n\}$ from the stripe set Θ according to the center position of the stripe in vertical, where n is the number of stripes of one image and all of the stripes in Θ_i have the same center position in vertical as h_i. Considering the slight vertical misalignment, we relax the above strict constraint to a larger space. Then,

$$\Theta_i = \{s_j | s_j \in \Theta, |h(s_j) - h_i| \leq h_\theta\}, \tag{1}$$

where $h(s)$ is the center position of the stripe s in vertical, h_θ is the size of relaxed space. In our setting, h_θ is equal to the stride for stripe sampling. When the stripe subsets are generated, we build stripe vocabulary trees for each stripe subset with the hierarchical k-means algorithm, denoted as $\{T_1, T_2, ..., T_n\}$. Then, those stripe vocabulary trees form a stripe vocabulary forest $F = \{T_1, T_2, ..., T_n\}$ and $sim(s, \Theta)$ can be estimated with it. Given a stripe s, when $h(s) = h_i$, then send s to T_i and s reaches the rth leaf node of T_i,

$$sim(s, \Theta) = \min_{s_j \in \tilde{S}_r^{T_i}} (d(s, s_j)), \tag{2}$$



(a) (b)

Fig. 5. (a) Illustration of the distance between different kinds of stripes and the stripe sets Θ^{P} and Θ^{N}. The distance is represented with the bar and the longer bar means the larger distance. S_1 is the stripe containing bag regions. S_2 is the stripe without bag regions. (b) Illustration of stripe feature extraction.

where $\tilde{S}_r^{T_i}$ is the stripe set associated with the rth leaf node of T_i and $d(s, s_j) = \left\|\mathbf{v}_s - \mathbf{v}_{s_j}\right\|_2$, $\mathbf{v}_x$ is the feature vector of the stripe x.

There are two advantages of adopting the stripe vocabulary forest. (1) $sim(s, \Theta)$ estimation is more accurate since it considers the particular structure of pedestrian and doesn't match the stripes in upper body to the ones in lower body. (2) It reduces the time consumption for estimating $sim(s, \Theta)$ due to $\Theta_i \subset \Theta$.

3.2 Classification score estimation

To recognize "has-bag", we firstly build two stripe vocabulary forests F^{P} and F^{N} with the stripe set Θ^{P} and Θ^{N}. Then learn a stripe classifier and estimate the likelihood of one stripe containing bag regions. The likelihood of stripe s_i can be represented as follows:

$$f(s_i) = \frac{sim(s_i, \Theta^{\mathrm{P}})}{sim(s_i, \Theta^{\mathrm{N}})} \tag{3}$$

As seen in Fig.5(a), when the stripe s_i contains bag regions, $sim(s_i, \Theta^{\mathrm{P}})$ will be small and $sim(s_i, \Theta^{\mathrm{N}})$ will be large, then $f(s_i)$ will be small. On the contrary, when the stripe s_i is without bag regions, because not all the stripes in bag images contain bag regions, both $sim(s_i, \Theta^{\mathrm{P}})$ and $sim(s_i, \Theta^{\mathrm{N}})$ will be small, then $f(s_i)$ will be large.

With the stripe classifier, we introduce a function $score(p)$ to check whether or not a pedestrian p is with a bag. This function can be represented as follows:

$$score(p) = \sum_{s_i \in S^p} f(s_i), \tag{4}$$

The final decision is made by assigning a confidence threshold.

3.3 Stripe feature extraction

A stripe is always represented by color and texture histogram features in person re-identification [5, 8, 9, 11], but those features ignore the spatial distribution and

Fig. 6. Some warped region proposals generated with selective search.

the neighborhood context of object. In our approach, we utilize contour features to describe a stripe. Fig.5(b) shows the procedure of stripe feature extraction. Given a stripe, we densely sample a set of 16×16 patches from this stripe and set the shift stride to 4 pixels both on vertical and horizontal direction. For each patch, we extract HOG feature [26] that is widely used to describe the contour information of object and of which gradient is voted into 9 orientation bins in $0° - 180°$. Then this stripe is depicted by concatenating HOG features of all patches.

4 Locate bag region

In this section, at first, we will show how to generate the region proposals. Then, we will present how to learn a region proposal ranker with a convolutional neural network.

4.1 Region proposals generation

In our approach, we use selective search [16] to generate the region proposals. Selective search exploits the structure of the image and generates object locations from super pixels. It uses a variety of complementary grouping criteria to diversify the sampling techniques and account for as many image conditions as possible. To be considered as a correct region proposal, the area of overlap a_o between the predicted bounding box B_p and the ground truth bounding box B_{gt} must exceed 50% by the formula:

$$a_o = \frac{area(B_p \cap B_{gt})}{area(B_p \cup B_{gt})}, \tag{5}$$

We observe that selective search will yield 96.04% recall and the average number of region proposals is only 463 for each pedestrian image containing bag in

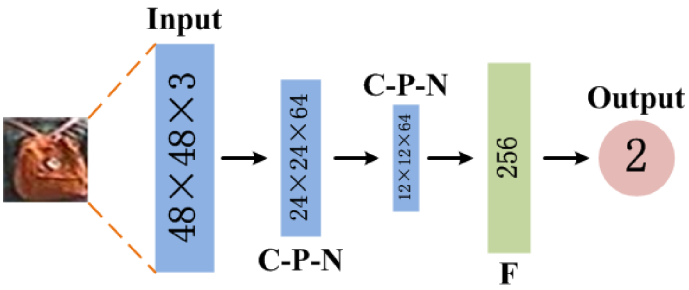


Fig. 7. Schematic view of the CNN model for ranking the region proposals. We visualize the network layers with their corresponding output dimensions.

CUHK dataset (the size of the images is 160×60). Thus, selective search is fit for generating the region proposals for bag detection in pedestrian images. To facilitate the following convolutional neural network design, we warp all region proposals in a fixed 48×48 pixel size. Fig.6 shows some warpped bag region proposals and some non-bag ones.

4.2 Region proposal ranking

To rank the region proposals, inspired from the classification on CIFAR-10 dataset with cuda-convnet [23], we consider the bag region proposals and the non-bag ones as two different classes and redesign a simple convolutional neural network. Fig.7 gives the schematic view of our CNN model. Denote by C a convolutional layer, by N a local response normalization one, by P a max pooling layer and by F a fully connected one. The network can be described concisely as follows: C $(48 \times 48 \times 64)$-P$(24 \times 24 \times 64)$-N$(24 \times 24 \times 64)$-C$(24 \times 24 \times 64)$-P$(12 \times 12 \times 64)$-N$(12 \times 12 \times 64)$-F(256). For C, P and N layers, the size is defined as width $\times$ height $\times$ depth, where the first two dimensions have a spatial meaning while the depth defines the number of filters. The input to the net is a 48×48 warped region proposal. The output layer of the net is a softmax layer with 2 output values that are the probabilities of a region proposal belonging to bag or not. The total number of parameters of the above CNN is about 2.5 million. For further details, we refer the reader to [23]. To train the above CNN model, we use only purely supervised learning approach as same as the procedure for training the traditional neural network.

5 Experiments

We utilize the pedestrian images from the public available dataset CUHK Person Re-identification Dataset [4] to evaluate our approach. The pedestrian images in this dataset are collected from the campus where the bag appearance and location change greatly, as seen in Fig.8. The size of these pedestrian images is

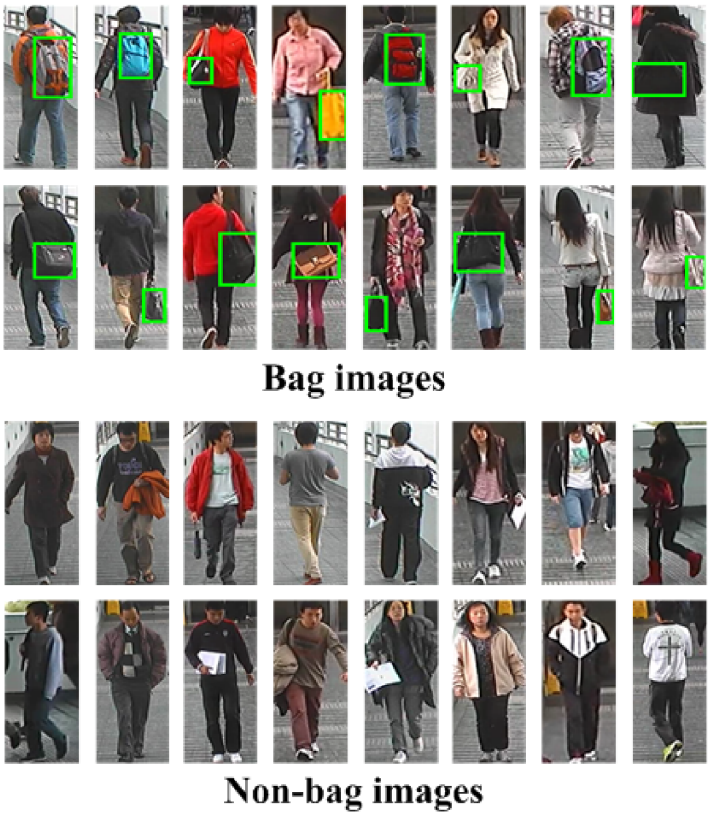
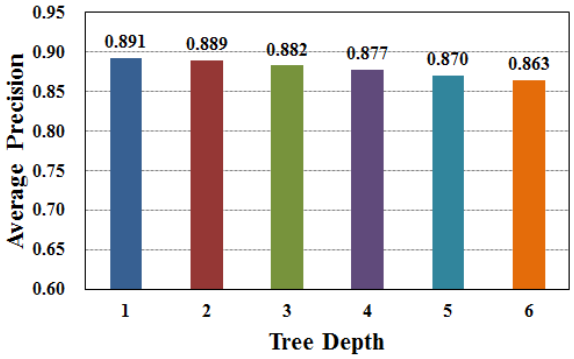


Fig. 8. Some bag images and some non-bag ones in CUHK dataset. The green boxes are ground truth of bag locations.

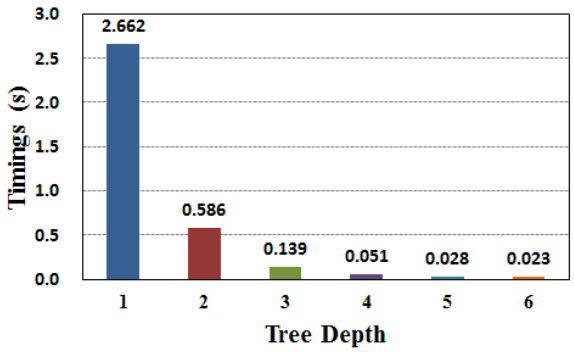
160×60 pixels and the type of bag includes handbag, backpack, briefcase, laptop bag, satchel, sling bag and hip bag.

To evaluate the performance of "has-bag" recognition, we manually label 1363 bag images and 1534 non-bag ones. Half of them are selected randomly for training and the other half for testing. To densely sample stripes from an image, the height of a stripe is set to be 32 pixels and the shift stride is set to be 4 pixels. Then the number of stripes of one image is 33. We use a branch factor $k = 5$ to train the stripe vocabulary forest. All experiments related to "has-bag" recognition were carried out with C++ implementation on a CPU.

To evaluate the performance of bag location, we manually label the ground truth bounding box of the above 1363 bag images. Half of them are selected randomly for training and the other half for testing. Because the number of bag region proposals is far less than non-bag ones, we adapt the following strategy to augment the amount of the bag region proposals. At first, we extend the 48×48 warped region proposals into 56×56 according to the original image. Then, we



(a) The average precision of precision-recall curve.



(b) The time consumption.

Fig. 9. Comparison of different tree depths for "has-bag" recognition.

densely sample a set of 48×48 region proposals from a 56×56 region proposal and set the shift stride to 1 pixels both at vertical and horizontal direction. Therefore, for training the CNN model, the number of the bag region proposals is about 3×10^5 and the non-bag ones are also about 3×10^5. All experiments related to bag location were carried out with Matlab and Python implementation on a GPU.

5.1 Evaluation on "has-bag" recognition

The tree depth exploration: The depth of the hierarchical k-means tree is a key parameter of our approach for "has-bag" recognition. Fig.9 compare the performance of different tree depths. When the tree depth is 1, it means that the similarity $sim(s, \Theta)$ between a stripe s and a stripe set Θ is estimated by 1-NN search and without the stripe vocabulary forests in our approach. When the tree depth is increased, although the average precision will be decreased, the time consumption will be reduced too. It is easy to observe that the average

Method	Average Precision	Timings (s)
SVF	**0.882**	**0.139**
SVT	0.871	1.237

Table 1. Comparisons of the stripe vocabulary forest (SVF) and the stripe vocabulary tree (SVT) for "has-bag" recognition.

precisions are comparable when the tree depth is changed from 1 to 3, but the time consumption is reduced obviously. When the tree depth is 3, it is about **20** times faster than the tree depth is 1. Although the time consumption is reduced further when the tree depth changes from 4 to 6, the average precisions are worse than the tree depth is 1. In the following sections, we will evaluate our approach for "has-bag" recognition under the tree depth is 3.

Forest vs. Tree: To estimate the similarity between a stripe and a stripe set, we utilize the stripe vocabulary forest instead of the stripe vocabulary tree constructed from all stripes in the stripe set. Table 1 compares the performance of our approach with the stripe vocabulary forest (SVF) and the stripe vocabulary tree (SVT). The average precision of SVF is about 1.1% higher than SVT and the time consumption of SVF is around 10 times faster than SVT for testing an image. SVF is superior to SVT.

Comparison with other approaches: To demonstrate the effectiveness of our approach for "has-bag" recognition, we will compare some other approaches that use features extracted from full body image. In those approaches, two kinds of feature vector are taken into account: histogram of oriented gradients (HOG) and histogram of color and texture (HCT). The HOG feature vector is similar as our stripe feature and is densely sampled a set of patches from an image. Then concatenate the HOG features of all patches to form the feature vector. The HCT feature vector is the same as [9, 11]. A pedestrian image is divided into six stripes equally. For each stripe, 8 color channels (RGB, YCbCr and HSV) and 21 texture filters (Schmid, Gabor) are used and each channel is described by a 16 dimensional histogram. Then, concatenate all histograms to form the feature vector. In order to recognize "has-bag", we train binary classifiers with KNN and SVM for each feature type separately. Learning KNN classifier is similar as our stripe classifier. Because the dimensionality of HOG feature vector is 15984 for a 160×60 image and it is large, the nonlinear mapping does not improve the performance [27], we train the SVM classifier by LibLinear [28]. For HCT feature vector, the dimensionality is 2784 for a 160×60 image and not large, we train the SVM classifier by LibSVM [27] and select RBF as the kernel function.

Fig.10 compares our approach with the other approaches for "has-bag" recognition. The average precision of our approach is about 2.9% higher than HOG-KNN, 4.6% higher than HOG-SVM, 23.7% higher than HCT-KNN and 15.1% higher than HCT-SVM. Our approach outperforms all the comparison approaches. Moreover, since concatenated HOG feature takes into account the spatial

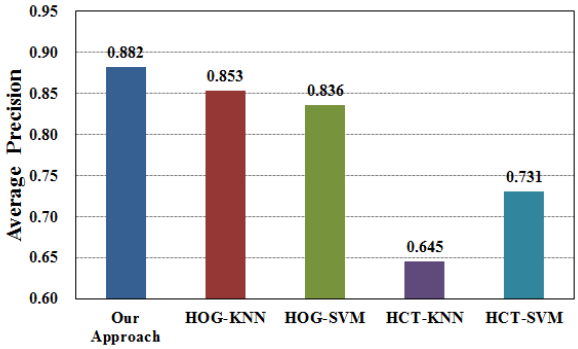


Fig. 10. Comparisons of our approach and the other approaches for "has-bag" recognition.

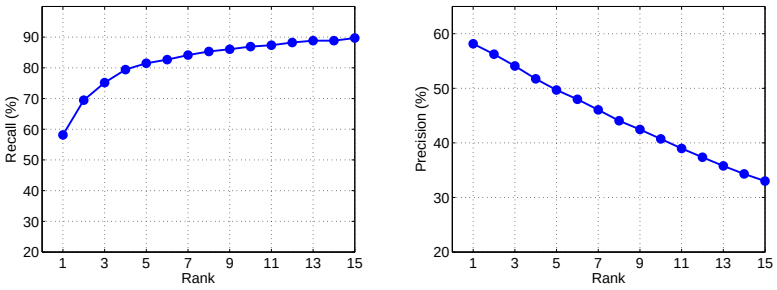
distribution and the neighborhood context of object, the performance of HOG feature is better than HCT feature.

5.2 Evaluation on bag location

Fig.11 shows the performance of our approach for bag location. When we regard the region proposals of each image in top 1 as the location results, both the precision and recall of our approach for bag location is 58.1%. Our approach can locate the bag regions effectively. Two points are worth highlighting concerning our bag location approach. (1) When we consider the region proposals in top 15, the recall is 89.7%. Our approach can also yield a high quality locations with a small number of the region proposals. (2) Assume that "Rank = r" denotes the region proposals of each image in top r are regarded as the location results. When Rank = 3, although the precision is 54.1% that is lower than Rank = 1, the recall is 75.2% that is much higher than Rank = 1. This indicates that we can utilize context information of the region proposals to further improve the bag location performance. Fig.12 shows some bag location results of our approach.

6 Conclusion

In this paper, we investigate bag information extraction in pedestrian images and attempt to tackle a practical visual surveillance problem of bag detection in pedestrian images. We propose a novel two-stage approach for this problem. At first, we check whether a pedestrian is with a bag using two stripe vocabulary forests. Then, we combine selective search and convolutional neural network to locate the bag region in the bag images. Experiments show that our approach is effective for bag detection in pedestrian images. In the future, we will utilize more bag images and non-bag ones and mine the context information of the region proposals to improve bag detection performance.



(a) Recall curve (b) Precision curve

Fig. 11. Recall and precision of the ranked region proposals. "Rank = r" denotes the region proposals of each image in top r are regarded as the location results.

Fig. 12. Some bag location results of our approach. The red boxes are ground truth of bag locations and the green ones are the detection results of our approach.

Acknowledgement. This work is supported in part by National Basic Research Program of China under Grant No.2011CB302203, and it is also supported by a grant from OMRON Corporation.

References

1. Zhao, R., Ouyang, W., Wang, X.: Unsupervised Salience Learning for Person Re-identification. In: CVPR. (2013)
2. Li, W., Wang, X.: Locally Aligned Feature Transforms across Views. In: CVPR. (2013)
3. Ma, B., Su, Y., Jurie, F.: BiCov: a Novel Image Representation for Person Re-identification and Face Verification. In: BMVC. (2012)
4. Li, W., Rui, Z., Wang, X.: Human Reidentification with Transferred Metric Learning. In: ACCV. (2012)
5. Zheng, W., Gong, S., Xiang, T.: Transfer Re-identification: From Person to Set-based Verification. In: CVPR. (2012)
6. Hirzer, M., Roth, P., Kostinger, M., Bischof, H.: Relaxed Pairwise Learned Metric for Person Re-identification. In: ECCV. (2012)

7. Wu, Y., Minoh, M., Mukunoki, M., Lao, S.: Set Based Discriminative Ranking for Recognition. In: ECCV. (2012)
8. Gray, D., Tao, H.: Viewpoint Invariant Pedestrian Recognition with an Ensemble of Localized Features. In: ECCV. (2008)
9. Zheng, W., Gong, S., Xiang, T.: Person Re-identification by Probabilistic Relative Distance Comparison. In: CVPR. (2011)
10. Satta, R., Fumera, G., Roli, F.: A General Method for Appearance-Based People Search Based on Textual Queries. In: Workshop of ECCV. (2012)
11. Layne, R., Hospedales, T., Gong, S.: Towards Person Identification and Re-identification with Attribute. In: Workshop of ECCV. (2012)
12. Damen, D., Hogg, D.: Detecting Carried Objects from Sequences of Walking Pedestrians. IEEE Trans. on PAMI **34** (2012) 1056–1067
13. Damen, D., Hogg, D.: Detecting Carried Objects in Short Video Sequences. In: ECCV. (2008)
14. BenAbdelkader, C., Davis, L.: Detection of People Carrying Objects: a Motion-based Recognition Approach. In: FG. (2002)
15. Haritaoglu, I., Cutler, R., Harwood, D., Davis, L.: Backpack: Detection of People Carrying Objects Using Silhouettes. In: ICCV. (1999)
16. Uijlings, J., Sande, K., Gevers, T., Smeulders, A.: Selective Search for Object Recognition. International Journal of Computer Vision **104** (2013) 154–171
17. Bourdev, L., Maji, S., Malik, J.: Describing People: A Poselet-Based Approach to Attribute Classification. In: ICCV. (2011)
18. Baltieri, D., Vezzani, R., Cucchiara, R.: People Orientation Recognition by Mixtures of Wrapped Distributions on Random Trees. In: ECCV. (2012)
19. Cao, L., Dikmen, M., Fu, Y., Huang, T.: Gender Recognition from Body. In: ACM MM. (2008)
20. Alexe, B., Deselaers, T., Ferrari, V.: Measuring the Objectness of Image Windows. IEEE Trans. on PAMI **34** (2012) 2189–2202
21. Endres, I., Hoiem, D.: Category Independent Object Proposals. In: ECCV. (2010)
22. Alex, K., Ilya, S., Geoffrey, H.: Imagenet Classification with Deep Convolutional Neural Networks. In: NIPS. (2012)
23. Alex, K.: Cuda-convnet. (https://code.google.com/p/cuda-convnet/)
24. Wang, X., Hua, G., Han, T.: Detection by Detections: Non-parametric Detector Adaptation for a Video. In: CVPR. (2012)
25. Nister, D., Stewenius, H.: Scalable Recognition with a Vocabulary Tree. In: CVPR. (2006)
26. Dalal, N., Triggs, B.: Histograms of Oriented Gradients for Human Detection. In: CVPR. (2005)
27. Hsu, C., Chang, C., Lin, C.: A Practical Guide to Support Vector Classification. In: Technical report, Department of Computer Science, National Taiwan University. (2003)
28. Fan, R., Chang, K., Hsieh, C., Wang, X., Lin, C.: LIBLINEAR: A Library for Large Linear Classification. In: JMLR **9** (2008) 1871–1874
29. Amer, M., Xie, D., Zhao, M., Todorovic, S., Zhu, S.: Cost-Sensitive Top-Down/Bottom-Up Inference for Multiscale Activity Recognition. In: ECCV. (2012)
30. Zhu, Y., Nayak, N., Roy-Chowdhury, A.: Context-Aware Modeling and Recognition of Activities in Video. In: CVPR. (2013)
31. Bhargava, M., Chen, C., Ryoo, M., Aggarwal, J.: Detection of Object Abandonment using Temporal Logic. Machine Vision and Applications **20** (2009) 271–281