Supervised Label Transfer for Semantic Segmentation of Street Scenes

Honghui Zhang[1], Jianxiong Xiao[2], and Long Quan[1]

[1] The Hong Kong University of Science and Technology
{honghui,quan}@cse.ust.hk
[2] Massachusetts Institute of Technology
jxiao@csail.mit.edu

Abstract. In this paper, we propose a robust supervised label transfer method for the semantic segmentation of street scenes. Given an input image of street scene, we first find multiple image sets from the training database consisting of images with annotation, each of which can cover all semantic categories in the input image. Then, we establish dense correspondence between the input image and each found image sets with a proposed KNN-MRF matching scheme. It is followed by a matching correspondences classification that tries to reduce the number of semantically incorrect correspondences with trained matching correspondences classification models for different categories. With those matching correspondences classified as semantically correct correspondences, we infer the confidence values of each super pixel belonging to different semantic categories, and integrate them and spatial smoothness constraint in a markov random field to segment the input image. Experiments on three datasets show our method outperforms the traditional learning based methods and the previous nonparametric label transfer method, for the semantic segmentation of street scenes.

1 Introduction

Semantic segmentation of street scenes is an important and interesting researching topic for scene understanding [1, 2] and image based modeling in cities and urban areas[3–6]. Traditional methods to solve this problem, such as [7–11], typically work with a fixed-number of object categories and train generative or discriminative models for each category. Recently, with the increasing availability of image collections with annotation, large database-driven approaches have shown the potential for nonparametric methods in several applications, such as object and scene recognition [12] and semantic segmentation [13]. Instead of training sophisticated parametric models, these methods try to reduce the inference problem for an unknown image to the problem of matching it to an existing set of annotated images by exploiting local similarity between images, which is addressed as label transfer in [13].

With scenes limited to street scenes, semantic segmentation is a suitable candidate for the application of the label transfer. Ideally, for an testing image, if

K. Daniilidis, P. Maragos, N. Paragios (Eds.): ECCV 2010, Part V, LNCS 6315, pp. 561–574, 2010.
© Springer-Verlag Berlin Heidelberg 2010

some high quality matches are contained in the training database with annotation, and we have a proper way to find them, the label transfer method [13] can work very well. However, in most cases, high quality matches are hardly available [14], even in street scenes, with exponential number of different object combinations within each scene. In addition, most existing methods for searching similar images are based on holistic similarity between images, such as widely used GIST descriptor [15]. They are computed on the layout of global content in images, which does not care about the quality of local matches. Without garantee of the local similarity between images, establishing semantically correct correspondence between images become very challenging. On the other hand, traditional methods [7–11] focus on local similarity for classifier training, and doesn't care about holistic image-level suitability.

This motivates us to investigate whether the combination of the traditional learning based methods and the pure label transfer can improve the performance of semantic segmentation of street scenes. In this paper, we propose a supervised label transfer method for semantic segmentation of street scenes, which introduces supervised classification models into the pure label transfer, to classify obtained matching correspondences and reject those untrusted matching correspondences.

The paper is structured as follows: a brief overview of our method is given in Section 2. In Section 3 and 4, the proposed KNN-MRF matching scheme and the supervised classification models for label transfer are introduced respectively. Then, the confidence inference, generation of segmentation mask for the input image and how to retrieve proper source for the supervised label transfer method are explained in Section 5. Finally, we evaluate and compare our method with related works in Section 6, and conclude in Section 7.

2 Overview

For a given input image, our method starts from finding some proper image sets with annotation from an existing database, each of which contain multiple images with annotation that cover all semantic categories in the input image. As mentioned above, it is difficult to find a single overall good match for query images. Some parts of the query image may be matched well, while some other parts could be totally missed. Based on this observation, it is claimed that a query image should be explained by a spatial composite of different regions taken from different images [14]. Inspired by this idea, we propose a new matching scheme for the label transfer that matches the given input image to each of the retrieved image sets, instead of matching it to a single image like [13].

To be specific, we perform a matching scheme that we call KNN-MRF matching between the input image and each of the retrieved image sets to establish a dense correspondence on super-pixel level. Then, it is followed by a matching correspondences classification step that uses some trained classification models to classify the matching correspondences and discard correspondences that are classified as semantically incorrect correspondences. Finally, with those matching correspondences that are classified as semantically correct correspondences,

we infer the confidence value of each basic matching element belonging to different categories, and then integrate these inferred confidence cues and spatial smoothness constraint into a markov random field to segment the input image. An outline of our method is given in Fig. 1. In the following section, we will first introduce the KNN-MRF matching scheme for an input image and an image set.

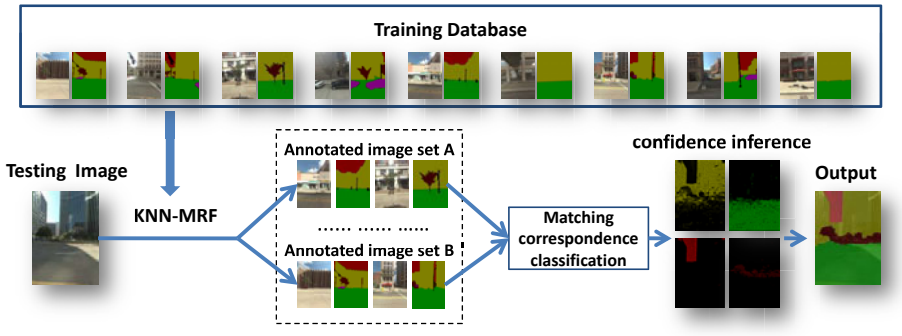


Fig. 1. Outline of our supervised label transfer algorithm

3 KNN-MRF Matching

In order to transfer labels of annotated images to an input image, we need to establish dense semantically correct correspondence between the input image and the annotated images. To make the matching process efficient, we use super pixel as basic matching element, instead of pixel as [13] did. As most super pixels are semantically coherent, using super pixel as basic matching element would be proper. For each image, a graph $\mathcal{G} = \langle \mathcal{V}, \mathcal{E} \rangle$ is defined, with each vertex $p \in \mathcal{V}$ in the graph denotes one super pixel in the image, while the edges $\mathcal{E}$ denotes the neighboring relationship between super pixels. Then we could formulate the matching problem as a simplified inexact graph matching problem [16]. More formally, given two graphs $\mathcal{G}_{\mathcal{I}} = (\mathcal{V}_{\mathcal{I}}, \mathcal{E}_{\mathcal{I}})$ and $\mathcal{G}_{\mathcal{A}} = (\mathcal{V}_{\mathcal{A}}, \mathcal{E}_{\mathcal{A}})$ that denote the input image and an retrieved image set for it respectively, the inexact graph matching problem consists in searching for a mapping from $\mathcal{V}_{\mathcal{I}} \rightarrow \mathcal{V}_{\mathcal{A}}$, with a constraint that each node in $\mathcal{V}_{\mathcal{I}}$ will be matched to another node in $\mathcal{V}_{\mathcal{A}}$. For the retrieved image set, we simply combine the graph for each image of it together to form a larger graph.

To solve this graph matching problem, we propose an efficient matching scheme that we call KNN-MRF matching scheme, named by the way we solve the problem. Given an input image I and an image set $\{A_i\}_{i=1}^{N}$ with annotation, we first find the K-Nearest-Neighbor(KNN) for each super pixel in I from $\{A_i\}_{i=1}^{N}$(in our implementation $K = 5$). Then we use a Markov random field built upon the graph $\mathcal{G}$ defined for I with the following energy function:

$$E(C) = \sum_{p_i \in \mathcal{V}} S(C_i) + \alpha \sum_{e_{ij} \in \mathcal{E}} f(C_i, C_j) \tag{1}$$

The candidate label set $\{C_i\}_{i=1}^{K}$for each super pixel in I consists of the index set $\{1, 2, ..., K\}$ for the corresponding K nearest neighbor from $\{A_i\}_{i=1}^{N}$, and $\mathcal{E}$ contains all the spatial neighborhood. In the energy function, $S(C_i)$ denotes the matching distance of each node to its C_i-th nearest neighbor. $f(C_i, C_j) = 0$ if the C_i-th nearest neighbor and C_j-th nearest neighbor of two neighboring super pixels in I are also neighboring, else $f(C_i, C_j) = 1$. For neighboring super pixels in I, by setting the smooth term with this way, matching to neighboring nearest neighbor will be given more preference. The alpha expansion algorithm [17] can be used to optimize the energy function and obtain a dense matching configuration. An example with detailed explanation is given in Fig. 2.

3.1 Superpixel Descriptor

Visual word has already been proven to be powerful in many visual problems, like object recognition and segmentation. We will combine it with super pixel to describe an image. For each image, it is first decomposed into many coherent regions, super pixels, by using the turbo pixel algorithm [18] which could make the size of super pixels balanced. Then the visual word descriptor for each super pixel is generated by quantizing features of pixels contained in the super pixel with a hierarchical k-mean clustering tree. In our implementation, we use the Texton feature [9] of pixels. To reduce time cost of searching the K nearest neighbor, for each image in the database, the visual word descriptor for each super pixel in it is precomputed and organized in a KD-tree for later reference.

3.2 Distance Metric

For the KNN-MRF matching between I and $\{A_i\}_{i=1}^{N}$, the distance metric used to retrieve the K-Nearest-Neighbor is defined as:

$$D(p, q) = \|(D_p - D_q)\|^2 + \beta(1 - [L_q \in R]) \tag{2}$$

where p and q are two super pixels in I and $\{A_i\}_{i=1}^{N}$ respectively, D_p and D_q are the feature descriptor of them. L_q is the label of q and known from the annotation of $\{A_i\}_{i=1}^{N}$, and R is the image level prior of the semantic categories contained in I. The image level prior is incorporated into the distance metric, and it has already been shown in [8], image level prior can improve semantic segmentation performance. When no image level prior is available, we set $\beta = 0$.

4 Matching Correspondences Classification

For the super-pixel matching correspondences obtained by the KNN-MRF matching between the input image I and the image set $\{A_i\}_{i=1}^{N}$ with annotation, we denote them as $\{\langle T_j, D_{T_j}, L_{T_j}, S_j, D_{S_j}, L_{S_j}\rangle\}$. T_j and S_j are the super pixels matched together in I and $\{A_i\}_{i=1}^{N}$ respectively, D_{T_j} and D_{S_j} are the descriptor for them, and L_{T_j} and L_{S_j} are the labels of them. To reduce the number

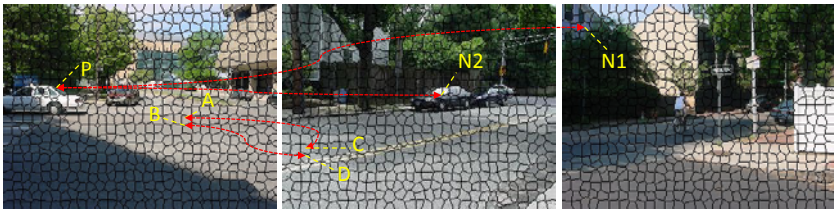


Fig. 2. KNN-MRF Matching between an input image (a) and an image set that consists of (b) and (c): for each super pixel P in (a), candidate label set for P in the energy function (1) consists of the K nearest neighbor of P, $\{N_i\}$ in (b) and (c); for neighboring super pixels A and B, matching to neighboring super pixel C and D will be given more preference by setting the smoothness term of the energy function (1) properly

of mismatch($L_{T_j} \neq L_{S_j}$), we train some matching correspondences classification models with the extremely randomized forest [19] to classify the obtained matching correspondences and discard those semantically incorrect matching correspondences, or mismatches.

For each matching correspondence $\langle T_j, D_{T_j}, L_{T_j}, S_j, D_{S_j}, L_{S_j}\rangle$, L_{S_j} is known from the annotation associated with $\{A_i\}_{i=1}^{N}$, so to distinguish whether it is a mismatch, we can reduce the problem to distinguish whether it is a mismatch for the certain category L_{S_j}. Therefore, instead of training a general classification model for all matching correspondences, we train a unary matching correspondences classification model for each category. The main advantage of training multiple matching correspondences classification models is improved performance, since certain cues and features are important for some categories and not for others.

To generate training samples for the matching correspondences classification model of a certain category L, we randomly select some image pairs $\{\langle A_m, A_n\rangle\}$ with annotation from the database, with both A_m and A_n containing L. Then for each super pixel in A_m, we find a nearest neighbor in A_n. By doing this, we can obtain many matching correspondences $\{\langle T_k, D_{T_k}, L_{T_k}, S_k, D_{S_k}, L_{S_k}\rangle\}$, where L_{T_k} and L_{S_k} have already been known. For a matching correspondence $\langle T_k, D_{T_k}, L_{T_k}, S_k, D_{S_k}, L_{S_k}\rangle$, it is taken as a positive training sample, if $L_{T_k} = L_{S_k} = L$, and a negative training sample if $L_{T_k} \neq L_{S_k}, L_{T_k} = L$ or $L_{T_k} \neq L_{S_k}, L_{S_k} = L$. In detail, given a correspondence $\langle T_k, D_{T_k}, L_{T_k}, S_k, D_{S_k}, L_{S_k}\rangle$, an appearance difference vector

$$V = |D_{T_k} - D_{S_k}|$$

combined with a position feature, the offset of their centers normalized with respect to the image width and height respectively.

$$\text{offset} = (|X_{T_k} - X_{S_k}|, |Y_{T_k} - Y_{S_k}|)$$

$\langle V, \text{offset}\rangle$ is used as the feature vector for the training samples of the matching correspondences classification model of category L.

The testing of a matching correspondence $\langle T, D_T, L_T, S, D_S, L_S \rangle$ is similar. Extract the feature vector $\langle V, \text{offset} \rangle$ first, then test it with the trained matching correspondences classification model for category L_S. If it is classified as a mismatch, we discard this matching correspondence.

5 Confidence Inference and MRF Integration

5.1 Selection of Proper Image Sets for Label Transfer

Given an input image, the first thing we need to do is selecting some proper image sets from the database, which can cover all semantic categories in the input image. We use the following way to find these image sets. First, we predict the image level prior R of the input image by retrieving the top K-nearest neighbor from the database with GIST matching [15], which has been proven a good way to predict the contents of images in [20]. Then we extract a subset V from the database: for each category $L \in R$, we retrieve the top K-nearest neighbor from the database with GIST matching and put them in V, with a constraint that they should also contain the category L. With this subset V, to generate an image set for matching, we randomly select some images from V until all categories in R have already been covered in at least one of the selected images. By repeating this process, we can get multiple image sets for the KNN-MRF matching.

5.2 Confidence Inference

With the multiple image sets obtained, we perform the KNN-MRF matching between the input image and each image set, followed by the matching correspondences classification. With matching correspondences$\{\langle T, D_T, L_T, S_n, D_{S_n}, L_{S_n} \rangle\}$ for each super pixel T in the input image, the confidence of T belonging to different categories are estimated as:

$$p(L_T = l|T) = \frac{F_T(l)W(l)}{\sum_{j=1}^{L} F_T(j)W(j)}, l = 1, 2, 3...K \tag{3}$$

where $F_T(l)$ is the occurrence of $\{\langle T, D_T, L_T, S_n, D_{S_n}, L_{S_n} : L_{S_n} = l \rangle\}$. In real situation, the frequency of occurrence of different category $\{P_k, k = 1, 2, ..., K\}$ could be quite different, which could make the matching bias toward categories with high frequency of occurrence. To overcome this problem, we introduce the weight term $W(l) = 1/P_k$ to balance the contribution of different categories.

5.3 Influence of Matching Correspondences Classification

In this part, we will analysis the influence of matching correspondences classification for the confidence inference. Suppose no matching correspondences

classification is done after the KNN-MRF matching, then the confidence of T belonging to different categories would be estimated as:

$$p'(L_T = l|T) = \frac{F'_T(l)W(l)}{\sum_{j=1}^{L} F'_T(j)W(j)}, l = 1, 2, 3...K \tag{4}$$

where $F'_T(l)$ is the occurrence of $\{\langle T, D_T, L_T, S_n, D_{S_n}, L_{S_n} : L_{S_n} = l\rangle\}$ obtained by KNN-MRF matching. Suppose the true label of T is i and the classification precision of each matching correspondences classification model is $p_k, k = 1, 2, 3...K$, so

$$E[F_T(i)] = p_i E[F'_T(i)] \tag{5}$$

$$E[F_T(j)] = (1 - p_j)E[F'_T(j)], i \neq j \tag{6}$$

where E denotes the mathematical expectation. With the matching correspondences classification integrated, we have

$$p(L_T = i|T) \approx \frac{p_i F'_T(i)W(i)}{p_i F'_T(i)W(i) + \sum_{j=1,j\neq i}^{L}(1 - p_j)F'_T(j)W(j)} \tag{7}$$

When the precision of each matching correspondences classification model is better than random guess, we have $p_k > 1/2, k = 1, 2, 3...K$. It is easy to prove that:

$$p'(L_T = i|T) < p(L_T = i|T) \tag{8}$$

At the same time, we have $\sum_{l=1}^{L} p'(L_T = l|T) = \sum_{l=1}^{L} p(L_T = l|T) = 1$, so it means the matching correspondences classification can enhance the contribution of correct matching correspondences in the confidence inference.

5.4 Markov Random Field Integration

Finally, we use a Markov random field to integrate the inferred confidence and spatial smoothness constraint to segment the input image. A graph $\mathcal{G} = \langle \mathcal{V}, \mathcal{E}\rangle$ is defined, with each vertex $p \in \mathcal{V}$ in the graph denotes one super pixel in the input image, while the edges $\mathcal{E}$ denotes the neighboring relationship between super pixels. Then we build a markov random field upon $\mathcal{G}$, with the energy function defined as:

$$E(\mathbf{L}) = \sum_{T\in\mathcal{V}} \psi_i(L_i) + \rho \sum_{e_{ij}\in\mathcal{E}} \psi_{ij}(L_i, L_j) \tag{9}$$

data term $\psi_i(L_i) = p(L_i|T)$ and smooth term

$$\psi_{ij}(i, j) = [L_i \neq L_j]\frac{1}{1 + \lambda \parallel D_i - D_j \parallel^2} \tag{10}$$

where D_i and D_j are the feature descriptor of two neighboring super pixels. The alpha expansion algorithm [17] can be used to optimize the energy function and obtain an optimal label configuration.

6 Experiments

We used three datasets to test our method: the Google street view dataset used in [10] ,the CBCL StreetScenes dataset [21] and the Cambridge-driving Labeled Video dataset(CamVid) [22]. They covered street scenes taken from different perspective under different lighting condition(day or dusk). As [8], we will use the category average accuracy (the average proportion of pixels correct in each category) and the global accuracy (total proportion of pixels correct) to evaluate the segmentation performance. For each dataset, 50% images by random selection are put in the database, and the left are used for testing. Our method does not require any 3D geometry information like [10, 11, 23], as it is not limited to the street scenes of image sequences or images of multiple view.

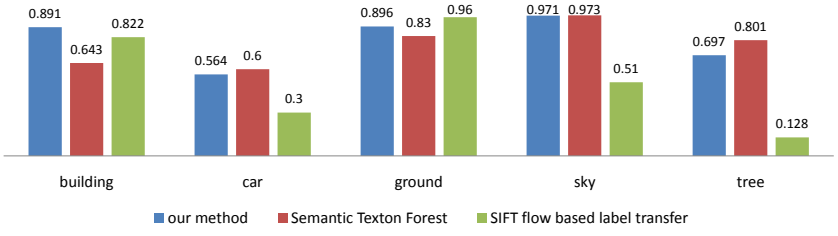
Parameter setting. For each dataset, we rescale images so that the average number of pixels contained in each image is about 320×240. For each input image, we select twenty annotated images sets for the KNN-MRF matching. When we decompose an image into super pixels with the method [18], the average size of super pixels is about one hundred pixels. For image level prior prediction and extracting the subset from the database introduced in section 5.1, the top five nearest neighbor by GIST matching are used.

6.1 Google Street View Dataset

This dataset consists of about 10,000 images captured in the downtown of Pittsburgh by Google Street View. For evaluation of our method, we randomly select 320 images from this dataset, and labeled them by hand with five categories: *sky*, *ground*, *building*, *car* and *tree*. 160 images are put in the database, and the left 160 images are used for testing. The evaluation includes the following two parts: comparison with two other methods, Semantic Texton Forest [8] and the SIFT flow based label transfer method [13]; and the influence of matching correspondences classification and the influence of different features in the matching correspondences classification models.

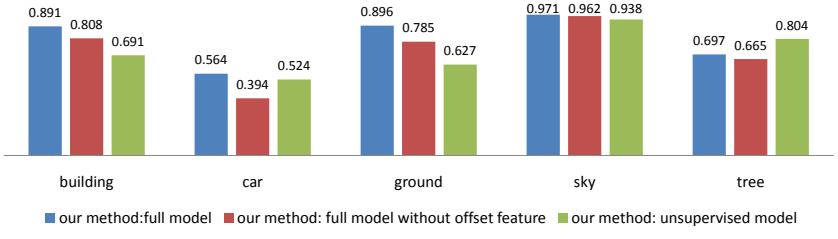
The comparison with [8, 13] is shown in Fig. 3(a). From the comparison result, we found that in terms of category average accuracy and global accuracy, our method is the best among the three methods. On the same dataset, the global accuracy reported in [10](without 3D geometry and spatial smoothness constraint) is 75.4%(In their evaluation, two more categories: *person* and *recycle bin* with frequency of occurrence under 1% are included).

To analysis how the matching correspondences classification and different features used in matching correspondences classification models influence the performance, we test our method with different setting: full model, full model without offset feature included in matching correspondences classification models and the unsupervised model without matching correspondences classification. The comparison is shown in Fig. 3(b). The comparison result shows that the matching correspondences classification brings significant performance improvement, in terms of category average accuracy and global accuracy. From the comparison of testing with full model and full model without offset feature included



	Our method	Semantic Texton Forest	SIFT flow based label transfer
Global accuracy	0.884	0.744	0.83
Category average accuracy	0.804	0.769	0.544

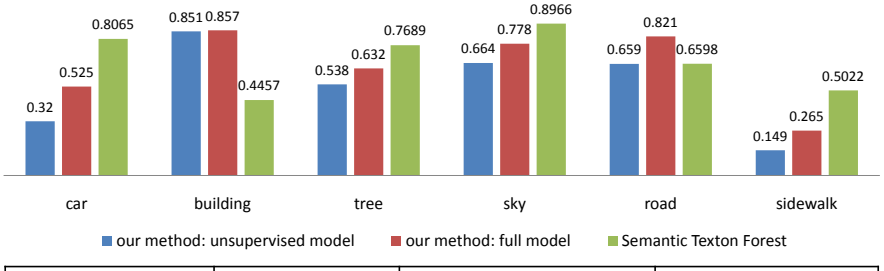
(a)



	full model	full model without offset feature	unsupervised model
Global accuracy	0.884	0.678	0.793
Category average accuracy	0.804	0.717	0.723

(b)

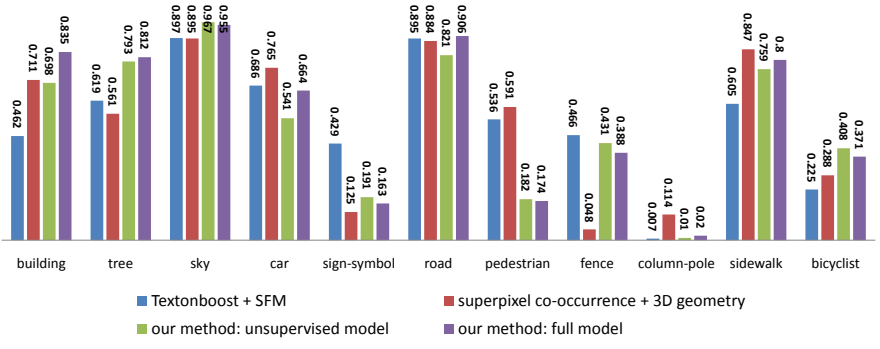
Fig. 3. (a) Segmentation accuracy of Our method, Semantic Texton Forest[8] and SIFT flow based label transfer[13] on the Google street view dataset; (b) Segmentation accuracy of our method with different setting on the Google street view dataset



	Our method: full model	Our method: unsupervised model	Semantic Texton Forest
Global accuracy	0.728	0.627	0.619
Category average accuracy	0.646	0.53	0.68

Fig. 4. Segmentation accuracy of our method and Semantic Texton Forest[8] on the CBCL StreetScenes dataset

in matching correspondences classification models, we found that the appearance feature and offset feature both contribute to the matching correspondences classification models. Some segmentation examples obtained by our method are given in Fig. 7(a).



	Our method: full model	Our method: unsupervised model	Textonboost + SFM	Superpixel co-occurrence + 3D geometry
Global accuracy	0.844	0.774	0.691	0.771
Category average accuracy	0.553	0.527	0.53	0.53

Fig. 5. Segmentation accuracy of our method, Textonboost + SFM [11] and Superpixel co-occurrence + 3D geometry [23] on the CamVid dataset

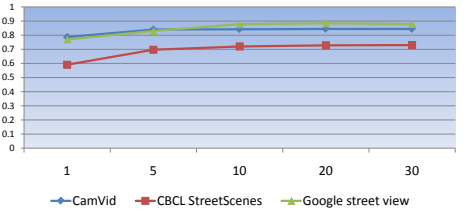
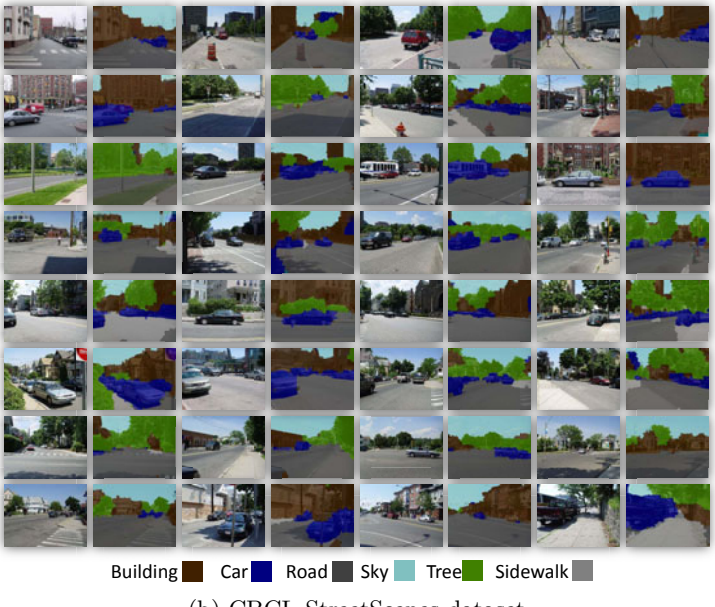


Fig. 6. Global accuracy(vertical axis) achieved by our method with different number of image sets(horizontal axis) used for confidence inference on different datasets

6.2 CBCL StreetScenes Dataset

The CBCL StreetScenes dataset contains total 3547 still images of street scenes with annotation, which mainly includes nine categories: *car*, *pedestrian*, *bicycle*, *building*, *tree*, *sky*, *road*, *sidewalk*, and *store*. In our test, the three categories with frequency of occurrence under 1%: *pedestrian*, *bicycle* and *store* are not included. We compared our method with one of the state-of-the-art semantic segmentation techniques: Semantic Texton Forest[8], and the comparison result is given in Fig. 4. The SIFT flow based label transfer method[13] is not included in the comparison, as the time cost of running it on the same train/test split is too high, over 800 hundred hours on a single computer by using the code the authors provided. In terms of category average accuracy, Semantic Texton Forest[8] is better than ours. However, the global accuracy of our method is about 11% higher than that of Semantic Texton Forest[8]. Same as that we found in the test on the previous dataset, the matching correspondences classification improved the performance of our method significantly. Some segmentation examples obtained by our method are given in Fig. 7(b).

(a) Google street view dataset



(b) CBCL StreetScenes dataset

Fig. 7. Some segmentation examples obtained by our method: (a) Google street view dataset;(b) CBCL StreetScenes dataset

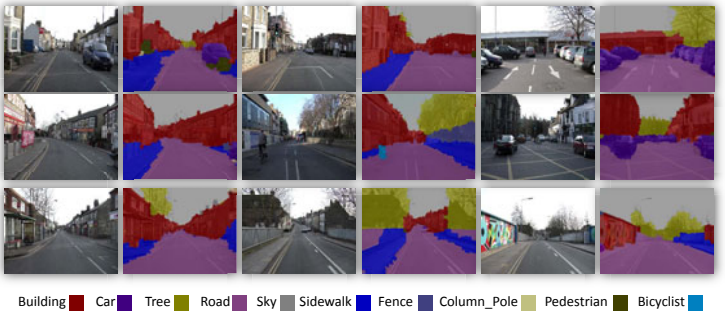


Fig. 8. Some segmentation examples obtained by our method on the CamVid dataset

6.3 CamVid Dataset

This dataset provides 701 still images with annotation under different lighting condition, which are extracted from video sequences. To compare with [11, 23] which used the same dataset for testing, we grouped the original label into 11 larger categories and downsampled the images to 320×240 pixels as [23]. The result obtained by our method and the result reported in [11, 23] is given in Fig. 5. From the comparison result we found that though no 3D geometry features are used in our method, our method still outperforms [11, 23]. At the same time, we found when no matching correspondences classification models are used, the performance of our method is still close to the state-of-the-art result reported [11, 23]. Last, the same as what we found in the testing on the previous two datasets, integrating matching correspondences classification into the label transfer brings a significant performance improvement. Some segmentation examples obtained by our method are given in Fig. 8.

6.4 Computation Time

The time cost of segmenting an input image depends on the size of the corresponding two graphs to be matched with the KNN-MRF matching and how many image sets are used for confidence inference. With our parameter setting, the average time cost for a single KNN-MRF matching between two graphs with average one thousand nodes is about one second. For all the three datasets, the average time cost to segment an input image with our method is under one minute. The global accuracy achieved by our method with different number of image sets used for confidence inference is given in Fig. 6.

7 Conclusion

We propose a supervised label transfer method for semantic segmentation of street scenes in this paper. Following the label transfer idea, given an input image, we first find multiple proper image sets from the database, each of which

can cover all semantic categories in the input image. Dense correspondence is established on super-pixel level between the input image and each found image sets with a proposed KNN-MRF matching scheme. Then it is followed by a matching correspondences classification that tries to reduce semantically incorrect correspondence with a supervised learning based method. With those matching correspondences classified as semantically correct correspondence, we infer the confidence value of each super pixel belonging to different semantic categories, and obtain the semantic segmentation of the input image by integrating the inferred confidence value and spatial smoothness constraint in a Markov random field. Experiments show encouraging performances on three standard datasets.

Acknowledgements. This work was partially supported by the Hong Kong RGC GRF 618908 and RGC GRF 619409, and the National Natural Science Foundation of China (60933006).

References

1. Bileschi, S.: StreetScenes: Towards Scene Understanding in Still Images. PhD thesis, Massachusetts Institute of Technology (2006)
2. Xiao, J., Hays, J., Ehinger, K., Oliva, A., Torralba, A.: SUN database: Large scale scene recognition from abbey to zoo. In: IEEE Conference on Computer Vision and Pattern Recognition (2010)
3. Xiao, J., Fang, T., Zhao, P., Lhuillier, M., Quan, L.: Image-based street-side city modeling. ACM Transactions on Graphics 28, 114:1–114:12 (2009)
4. Zhao, P., Fang, T., Xiao, J., Zhang, H., Zhao, Q., Quan, L.: Rectilinear parsing of architecture in urban environment. In: IEEE Conference on Computer Vision and Pattern Recognition (2010)
5. Xiao, J., Fang, T., Tan, P., Zhao, P., Ofek, E., Quan, L.: Image-based façade modeling. ACM Transactions on Graphics 27, 161:1–161:10 (2008)
6. Tan, P., Fang, T., Xiao, J., Zhao, P., Quan, L.: Single image tree modeling. ACM Transactions on Graphics 27, 108:1–108:7 (2008)
7. He, X., Zemel, R., Carreira-Perpinan, M.: Multiscale conditional random fields for image labeling. In: IEEE Conference on Computer Vision and Pattern Recognition (2004)
8. Shotton, J., Johnson, M., Cipolla, R.: Semantic texton forests for image categorization and segmentation. In: IEEE Conference on Computer Vision and Pattern Recognition (2008)
9. Shotton, J., Winn, J., Rother, C., Criminisi, A.: TextonBoost for image understanding: Multi-Class object recognition and segmentation by jointly modeling texture, layout, and context. International Journal of Computer Vision 81, 2–23 (2009)
10. Xiao, J., Quan, L.: Multiple view semantic segmentation for street view images. In: IEEE International Conference on Computer Vision (2009)
11. Brostow, G., Shotton, J., Fauqueur, J., Cipolla, R.: Segmentation and recognition using structure from motion point clouds. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) ECCV 2008, Part I. LNCS, vol. 5302, pp. 44–57. Springer, Heidelberg (2008)

12. Torralba, A., Fergus, R., Freeman, W.: 80 million tiny images: a large dataset for non-parametric object and scene recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence 30(11), 1958–1970 (2008)
13. Liu, C., Yuen, J., Torralba, A.: Nonparametric scene parsing: label transfer via dense scene alignment. In: IEEE Conference on Computer Vision and Pattern Recognition (2009)
14. Russell, B., Efros, A., Sivic, J., Freeman, W., Zisserman, A.: Segmenting scenes by matching image composites. In: Advances in Neural Information Processing Systems (2009)
15. Oliva, A., Torralba, A.: Modeling the shape of the scene:a holistic representation of the spatial envelope. International Journal of Computer Vision 42, 145–175 (2001)
16. Bengoetxea, E.: Inexact Graph Matching Using Estimation of Distribution Algorithms. PhD thesis, Ecole Nationale Supérieure des Télécommunications (2002)
17. Kolmogorov, V., Zabih, R.: What energy functions can be minimized via graph cuts? In: Heyden, A., Sparr, G., Nielsen, M., Johansen, P. (eds.) ECCV 2002. LNCS, vol. 2352, pp. 65–81. Springer, Heidelberg (2002)
18. Levinshtein, A., Stere, A., Kutulakos, K., Fleet, D., Dickinson, S., Siddiqi, K.: Turbopixels: Fast superpixels using geometric flows. IEEE Transactions on Pattern Analysis and Machine Intelligence 31, 2290–2297 (2009)
19. Moosmann, F., Triggs, B., Jurie, F.: Fast discriminative visual codebooks using randomized clustering forests. In: Advances in Neural Information Processing Systems (2006)
20. Russell, B., Torralba, A., Liu, C., Fergus, R., Freeman, W.: Object recognition by scene alignment. In: Object Recognition by Scene Alignment. Advances in Neural Information Processing Systems (2007)
21. Bileschi, S.: CBCL streetscenes challenge framework (2007), `http://cbcl.mit.edu/software-datasets/streetscenes/`
22. Brostow, G., Fauqueur, J., Cipolla, R.: Semantic object classes in video: A high-definition ground truth database. Pattern Recognition Letters 30(2), 88–97 (2009)
23. Micusik, B., Kosecka, J.: Semantic segmentation of street scenes by superpixel co-occurrence and 3d geometry. In: IEEE Workshop on Video-Oriented Object and Event Classification (VOEC) (2009)