Cross-domain Traffic Scene Understanding: A Dense Correspondence based Transfer Learning Approach

Shuai Di, Honggang Zhang, *Senior Member, IEEE,* Chun-Guang Li *Member, IEEE*, Xue Mei, *Senior Member, IEEE,* Danil Prokhorov, *Senior Member, IEEE,* and Haibin Ling

***Abstract*—Understanding traffic scene images taken from vehicle-mounted cameras is important for high level tasks such as Advanced Driver Assistance Systems (ADASs) and autonomous driving. It is a challenging problem due to large variations under different weather or illumination conditions. In this paper, we tackle the problem of traffic scene understanding from a cross-domain perspective. Specifically, we attempt to understand the traffic scene from images taken from the same location but under different weather or illumination conditions (e.g. understanding the same traffic scene from images on a rainy night with the help of images taken on a sunny day). To this end, we propose a Dense Correspondence based Transfer Learning (DCTL) approach, which consists of three main steps: a) extracting deep representations of traffic scene images via a Convolutional Neural Network (CNN), b) constructing compact and effective representations via cross-domain metric learning and subspace alignment for cross-domain retrieval, and c) transferring the annotations from the retrieved best matching image to the test image based on cross-domain dense correspondences and a probabilistic Markov random field (MRF). To verify the effectiveness of our DCTL approach, we conduct extensive experiments on a challenging data set, which contains 1,828 images from six weather or illumination conditions.**

***Index Terms*—Traffic scene understanding, semantic segmentation, transfer learning, dense correspondence, road scene, vehicle environment perception.**

I. Introduction

Understanding traffic scenes from images taken by vehicle-mounted cameras is important for situational awareness in Intelligent Transportation System (ITS), such as Advanced Driver Assistance Systems (ADAS) and autonomous driving. The state of art has mainly focused on road related detections, such as road layout detection [1], [2] and road marking detection [3], [4], [5], [6]. It is well accepted that a practical autonomous driving system requires reliable and effective traffic scene understanding [7], [8], [9], [10]. Existing approaches in traffic scene understanding, however, are sensitive to the large variations due to weather or illumination changes.

In this paper, we address the problem from a cross-domain transfer learning perspective, i.e. addressing it by using images of the same location but taken in other weather or illumination conditions. We assume that the annotated training images under good weather or illumination conditions are available for our reference. We view different weather or illumination conditions as different domains. Therefore, our problem is effectively a cross-domain learning problem.

Our basic idea is to find a subset of well-annotated images in good weather or illumination conditions and then transfer their annotations to the test image. Specifically, we propose a Dense Correspondence based Transfer Learning (DCTL) approach, in which we construct compact representations for finding the best matching image in a training set across domains, and then infer the annotations of the test image by building cross-domain dense correspondences between the test image and the retrieved best matching image in the training set.

In our proposed DCTL approach, we fine-tune a pre-trained Convolutional Neural Network (CNN) to extract deep representations of traffic scene images at first, and then perform domain adaptation to construct compact and effective representations for retrieving the best matching image in the training set across different domains (i.e. weather or illumination conditions). Finally, cross-domain dense correspondences between the test image and the best matching image with annotations are built via SIFT flow, and the annotations from the training images are transferred to the test image via a probabilistic MRF model. We verify effectiveness of our proposed approach on a challenging data set.

We summarize contributions of our paper below:

- We propose a dense correspondence based transfer learning framework for understanding traffic scene images under challenging variations of weather or illumination conditions. To the best of our knowledge, this is the first time that the traffic scene understanding is approached by transferring information from images of the same location taken in different conditions.
- We evaluate performance of state-of-the-art CNNs with different architectures, pre-trained on different data sets, with their deep features extracted from different layers.
- We collect samples to build a challenging image data set, which contains 1,828 images from six weather or illumination conditions. This data set is available online

S. Di is with the School of Information and Communication Engineering, Beijing University of Posts and Telecommunications, Beijing 100876, China (e-mail: renjie130@gmail.com).

H. Zhang and C.-G. Li are with the School of Information and Communication Engineering, Beijing University of Posts and Telecommunications, Beijing 100876, China (e-mail: {zhhg, lichunguang}@bupt.edu.cn).

X. Mei and D. Prokhorov are with the Toyota Research Institute, North America, Ann Arbor, MI 48105 USA (e-mail: nathanmei@gmail.com; dvprokhorov@gmail.com).

H. Ling is with the Department of Computer and Information Sciences, Temple University, Philadelphia, PA 19122 USA (e-mail: hbling@temple.edu). Corresponding author.

Manuscript received XXX, XX, 20XX; revised XXX, XX, 20XX.

for free to use for ITS research.

The remainder of this paper is arranged as follows. In Section II, we review the related work. In Section III, we present our DCTL approach. In Section IV, we describe our extensive experiments, followed by our conclusions in Section V.

II. Related Works

Our cross-domain transfer learning approach for traffic scene understanding is related to place recognition, domain adaptation, and scene recognition and semantic segmentation with deep learning.

A. Place Recognition

In the past few years, place recognition has achieved great progress [11], [12], [13], [14], [15], [16]. Roughly, the task of place recognition is treated as a variant of image retrieval problem [17]. The state-of-the-art approaches for place recognition are based on local invariant features, including image-level descriptors [16], [12] or reconstructed 3D points [15], [14]. In [13], Milford et al. introduce a condition-invariant method for place recognition, in which the images of the same location are matched and the highly aliased images from different locations are rejected. However, there is a lack of methods for understanding traffic scene images under different weather or illumination conditions.

B. Subspace-based Domain Adaptation

The idea of subspace-based domain adaptation is to project both source data and target data into a common subspace to make the distributions of the two sources as consistent as possible [18], [19], [20], [21]. We assume that there are many labeled data in the source domain but few in the target domain. We aim at adapting information from the labeled data in the source domain to the new data in the target domain.

In traffic scene understanding, we view the weather or illumination conditions as domains, and we treat our problem as cross-domain learning. We address the problem of recognizing images of the same scene across domains by learning a transform utilizing the data from two domains, and the domains may have large appearance variations.

C. Deep Learning for Scene Recognition/Semantic Segmentation

CNN based models have been the top performers on scene recognition tasks [22], [23], [24], [25]. In recent works [26], [27], [28], [29], deep CNN features learned on large data sets, such as ImageNet (ILSVRC) [30] and Places [23], [31], can be used as powerful descriptors to other applications.

However, traffic scene images considered in this paper have significant appearance variations. This differs from images in other data sets, such as those used for training ImageNet (ILSVRC) and Places which might not adequate for dealing with traffic scene images.

Deep architectures designed for semantic scene segmentation have also achieved the state-of-the-art results by learning to decode low resolution image representations to pixel-wise predictions such as [32], [33], [34], [35], [36]. The performance of these methods may degenerate if the images have large appearance variations as in our problem. Different from training a deep network directly, we perform our semantic scene segmentation by building dense correspondences between a test image and annotated images of the training set.

TABLE I
Pair-wise domains for the cross-domain traffic scene dataset.

High contrast domains:	
sunny day⟶ night	sunny day⟶ rainy night
cloudy day⟶ night	cloudy day⟶ rainy night
snowy day⟶ night	snowy day⟶ rainy night
Low contrast domains:	
sunny day⟶ foggy day	sunny day⟶ snowy day
cloudy day⟶ snowy day	rainy night⟶ night

III. Our Proposal: Dense Correspondence based Transfer Learning Approach

We describe our problem setting, followed by our proposed approach.

Problem Settings: We consider the following six typical weather and illumination conditions: *sunny day, night, snowy day, rainy night, cloudy day and foggy day*, with each condition viewed as a specific domain. Each domain contains traffic scene images taken at different locations, and each location is selected as one class. In addition, pair-wise domains are assembled and divided into two groups in terms of their illumination contrast: low contrast domains and high contrast domains as shown in Table I. The symbol "⟶" in the table points from the source domain to the target domain, and it means that we want to understand the traffic scene images in the target domain by transferring information from images in the source domain. We also select very challenging scenarios, e.g. *night* and *rainy night* as the target domains, with other scenarios as the source domains.

We illustrate the flowchart of our proposed approach DCTL in Fig. 1. The approach consists of three stages:

- Extracting deep features via a fine-tuned CNN;
- Constructing compact and effective representations via cross-domain metric learning and subspace alignment for cross-domain retrieval;
- Building cross-domain dense correspondences for transferring annotations.

A. Extracting Deep Representation

As shown in [26], [27], [28], [29], [37], a well-trained CNN can be used to generate powerful descriptors for applications on diverse data sets. Different CNN architectures have been proposed recently, e.g. VGG [28], [24], GoogLeNet [25], and DeCAF [26], [38]. We compare performance of their deep representations on traffic scene images.

As shown in [38], [28], fine-tuning the pre-trained CNN on a specific data set can improve the performance significantly. In our case, image appearances from our traffic scene data

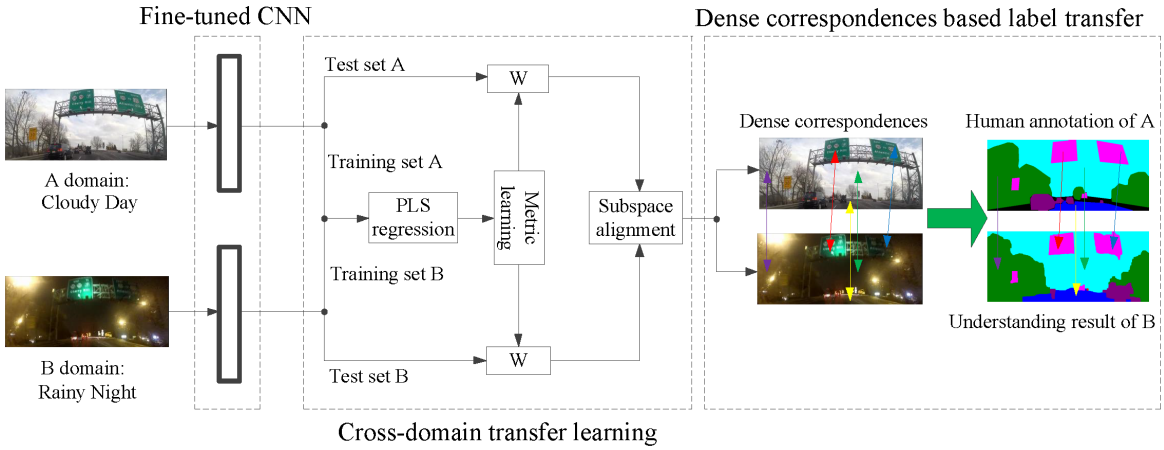


Fig. 1. Flowchart of the cross-domain traffic scene understanding.

sets are quite different from those in data sets used to pre-train CNN. That is why we fine-tune the pre-trained CNN on our traffic scene data sets.

A CNN often contains a huge number of adjustable parameters, e.g. more than 60 million parameters in the CNN architecture from [22]. Learning effectively so many parameters using images from modest-size data sets is infeasible. As shown in [26], [27], [39], the internal layers of the CNN can act as a generic extractor of image representations. Parameters of the internal layers of the pre-trained networks can remain unchanged before fine-tuning. In addition, data augmentation is applied, which is used to enlarge the data set artificially using label-preserving transformations [22], [27]. More specifically, we combine the horizonal reflections with crops, which is similar to recent data augmentation methods for training CNN [22], [27], [28]. In our data augmentation, ten samples are produced for each original image.

B. Domain Adaptation and Subspace Alignment

Traffic scene images taken in different weather or illumination conditions may have dramatic appearance variations, and the extracted deep features may also be exhibiting large feature variations. We tackle this difficulty by domain adaptation. As listed in Table I, the condition on the left side of the arrow is domain A and the condition on the right side of the arrow is domain B. For example, "sunny day", "cloudy day", and "snowy day" are domain A; whereas "night", "rainy night", and "foggy day" are domain B.

Our idea is to transfer the annotation information of traffic scene images in domain A to the test image in domain B. To do so, we need to find in the training set of domain A a subset of images, which are the best match to the test image. We then transfer the annotations by building dense correspondences to be described in the next subsection.

1) Training Stage: PLS Regression and Cross-Domain Metric Learning: Generally, PCA is the most popular method for linear dimension reduction before conducting metric learning. However, PCA is not able to preserve the latent structure across different domains as in our case. Therefore, instead, we apply PLS regression [40] on data from the two domains to learn compact representations of a common subspace.

Let training data in domain A and domain B be $X^{(a)} = [\mathbf{x}_1^{(a)}, \ldots, \mathbf{x}_n^{(a)}]$ and $X^{(b)} = [\mathbf{x}_1^{(b)}, \ldots, \mathbf{x}_m^{(b)}]$, which contain n and m deep features of d-dimension, respectively. Moreover, we arrange the labels of the training samples in domain A and domain B into label matrices $Y^{(a)} = [y_1^{(a)}, \ldots, y_n^{(a)}]$ and $Y^{(b)} = [y_1^{(b)}, \ldots, y_m^{(b)}]$, respectively, in which the labels $y_i^{(a)}$ and $y_j^{(b)}$ indicate the specific locations where the training samples are taken. If $y_i^{(a)} = y_j^{(b)}$, we call the paired samples $(\mathbf{x}_i^{(a)}, \mathbf{x}_j^{(b)})$ as a positive sample pair, otherwise we call it as a negative sample pair.

PLS regression is applied to the training data $\{X^{(a)}, X^{(b)}\}$, to obtain the projection matrix P of $d \times p$, where $p < d$ is the target dimension.

We denote the PLS dimension-reduced data as $\tilde{X}^{(a)}$ and $\tilde{X}^{(b)}$, where $\tilde{X}^{(a)} = P^T X^{(a)}$ and $\tilde{X}^{(b)} = P^T X^{(b)}$. Then, we learn a metric to measure the cross-domain distance between data samples from the two domains:

$$\|\tilde{\mathbf{x}}^{(a)} - \tilde{\mathbf{x}}^{(b)})\|_W^2 = (\tilde{\mathbf{x}}^{(a)} - \tilde{\mathbf{x}}^{(b)})^T W (\tilde{\mathbf{x}}^{(a)} - \tilde{\mathbf{x}}^{(b)}), \tag{1}$$

where W is a positive semi-definite matrix of $p \times p$.

Let $W = VV^T$ in which $V \in \mathbb{R}^{p \times q}$ with $q < p$, we have that:

$$\|\tilde{\mathbf{x}}^{(a)} - \tilde{\mathbf{x}}^{(b)})\|_W^2 = \|V^T \tilde{\mathbf{x}}^{(a)} - V^T \tilde{\mathbf{x}}^{(b)}\|_2^2, \tag{2}$$

Similar to [41], we use the log-logistic loss function as follows:

$$\ell_W(\tilde{\mathbf{x}}_i^{(a)}, \tilde{\mathbf{x}}_j^{(b)}) = \log(1 + e^{\theta_{ij}(\|\tilde{\mathbf{x}}_i^{(a)} - \tilde{\mathbf{x}}_j^{(b)}\|_W^2 - c)}), \tag{3}$$

where $\theta_{ij} = 1$ if $y_i^{(a)} = y_j^{(b)}$ and otherwise $\theta_{ij} = -1$, c is a constant. Then, by using $W = VV^T$, our cross-domain metric learning problem is formulated as follows:

$$\min_V \sum_{i=1}^{n} \sum_{j=1}^{m} \alpha_{ij} \ell_{VV^T}(\tilde{\mathbf{x}}_i^{(a)}, \tilde{\mathbf{x}}_j^{(b)}), \tag{4}$$

where $\alpha_{ij} = \frac{1}{N_+}$ if $\theta_{ij} = 1$ and $\frac{1}{N_-}$ otherwise, and N_+ and N_- are the numbers of positive and negative sample pairs, respectively. Note that the weighting scheme is important because N_+ and N_- are heavily unbalanced in our problem.

We solve problem (4) by the accelerated proximal gradient optimization method as in [41]. After the optimization process completion, we obtain the optimal solution V_* and apply it to find a compact and effective representation of the test data.

2) Testing Stage: Subspace Alignment: In the training stage, we perform PLS regression and cross-domain metric learning on the training data to minimize the discrepancy in the two domains. We obtain a latent structure preserving projection matrix P and a supervised domain adaptation projection matrix V_*.

In the testing stage, we apply the cross-domain projections P and V_* to reduce the dimensionality of the test data as follows:

$$\tilde{Z}^{(a)} = V_*^T P^T Z^{(a)}, \tag{5}$$

$$\tilde{Z}^{(b)} = V_*^T P^T Z^{(b)}, \tag{6}$$

where $Z^{(a)} = [\mathbf{z}_1^{(a)}, \ldots, \mathbf{z}_s^{(a)}]$ and $Z^{(b)} = [\mathbf{z}_1^{(b)}, \ldots, \mathbf{z}_t^{(b)}]$ contain s and t testing samples of d-dimension in domain A and domain B, respectively.

To further minimize the discrepancy between the two domains, we align subspace $\tilde{Z}^{(a)}$ in domain A with respect to subspace $\tilde{Z}^{(b)}$ in domain B. Let $Q_{(a)} \in \mathbb{R}^{q\times k}$ and $Q_{(b)} \in \mathbb{R}^{q\times k}$ be the left singular matrices of $\tilde{Z}^{(a)}$ and $\tilde{Z}^{(b)}$, respectively, then we can find an alignment matrix R by minimizing the Bregman matrix divergence [18] as follows:

$$\min_R \|Q_{(a)}R - Q_{(b)}\|_F^2, \tag{7}$$

where $\|\cdot\|_F$ is the Frobenius norm. Note that the closed-form solution is $R_* = Q_{(a)}^T Q_{(b)}$. Then, $\tilde{Z}^{(a)}$ and $\tilde{Z}^{(b)}$ can be projected into a common subspace as follows:

$$C^{(a)} = R_*^T Q_{(a)}^T \tilde{Z}^{(a)}, \tag{8}$$

$$C^{(b)} = R_*^T Q_{(b)}^T \tilde{Z}^{(b)}, \tag{9}$$

where $C^{(a)} \in \mathbb{R}^{k\times s}$ and $C^{(b)} \in \mathbb{R}^{k\times t}$ are the compact cross-domain representations for test data in domain A and domain B, respectively. For each test sample in domain B, we use its k-dimensional representation to find the best matching samples in domain A.

Remark. In [42], a metric learning is used to generate subspaces of the source domain. Unlike [42], we learn a metric on the cross-domain training data (resp. just one scenario data) and transfer the data which are different to the data used for learning the metric (resp. the same data used for learning the metric) into the metric-induced space.

C. Scene Understanding through Label Transfer

While the images in different domains are usually of different appearances due to variations from weather or illumination conditions, they share similar spatial layout structure. Therefore, the annotation information on scene images in domain A can be transferred into scene images in domain B if correct correspondences are properly created. In this paper, we build the dense correspondences via SIFT flow [43], [44] and transfer the annotation information via a Markov random field model.

1) Cross-domain Dense Correspondence via SIFT Flow: The goal of SIFT flow is to find the dense correspondences between two images. We consider a test image, denoted as $I^{(b)}$ and the best matching image, denoted as $I^{(a)}$. Let p be the spatial coordinates of a pixel in the image, and $\mathbf{f}^{(b)}(p)$ be the SIFT descriptor [45] at coordinates p in the test image $I^{(b)}$, $\mathbf{f}^{(a)}(p)$ be the SIFT descriptor at coordinates p in the best matching image $I^{(a)}$, and $w(p)$ be the displacement of the corresponding SIFT feature in image $I^{(a)}$. Similar to [46], we define the energy function of SIFT flow field[1] $\mathcal{W}$ on the best matching image $I^{(a)}$ with respect to test image $I^{(b)}$ as follows:

$$\begin{aligned}\varepsilon(\mathcal{W}) = &\sum_p \|\mathbf{f}^{(b)}(p) - \mathbf{f}^{(a)}(p + w(p))\|_2 \\ &+ \lambda \sum_{(p,q)\in\mathcal{E}} \|w(p) - w(q)\|_2^2,\end{aligned} \tag{10}$$

where $\mathcal{E}$ contains all of the spatial neighborhood (4-neighbor graph) and λ is the regularization parameter. We solve for $\mathcal{W}$ by minimizing the energy function $\varepsilon(\mathcal{W})$ using belief propagation [47].

2) Annotation Transfer: Given a test image $I^{(b)}$ with its corresponding SIFT descriptor field [2] $\mathcal{F}(I^{(b)})$, the best matching image $I^{(a)}$ with its corresponding SIFT descriptor field $\mathcal{F}(I^{(a)})$ and annotation field[3] $\mathcal{L}(I^{(a)})$, and the SIFT flow field $\mathcal{W}_*$ obtained from solving (10), our goal is to infer the annotation defined for each pixel in $I^{(b)}$, i.e. the annotation field $\mathcal{L}(I^{(b)})$ for test image $I^{(b)}$.

To infer the annotation field $\mathcal{L}(I^{(b)})$, we consider the dense correspondences between $I^{(b)}$ and $I^{(a)}$, spatial layout prior information on $I^{(b)}$ and the spatial smoothness in $I^{(b)}$.

To utilize the dense correspondences established by SIFT flow similar to [48], we define a penalty term $\phi(\mathcal{L}(I^{(b)}, p))$ as:

- If $\mathcal{L}(I^{(b)}, p) = \mathcal{L}(I^{(b)}, p + w_*(p))$, then
$$\phi(\mathcal{L}(I^{(b)}, p)) = \|\mathbf{f}^{(b)}(p) - \mathbf{f}^{(a)}(p + w_*(p))\|_2. \tag{11}$$
- If $\mathcal{L}(I^{(b)}, p) \neq \mathcal{L}(I^{(b)}, p + w_*(p))$, then
$$\phi(\mathcal{L}(I^{(b)}, p)) = \chi, \tag{12}$$
where $\mathcal{L}(I^{(b)}, p)$ is the annotation at position p in image $I^{(b)}$, and χ is sufficiently large, e.g. $\chi = \max_{p,q} \|\mathbf{f}^{(b)}(p) - \mathbf{f}^{(a)}(q)\|_2$.

To utilize our spatial prior, we define the penalty term:

$$\theta(\mathcal{L}(I^{(b)}, p)) = -\log \mathcal{H}(p), \tag{13}$$

where $\mathcal{H}(p)$ denotes the prior probability of pixel p to belong to an object category, and it can be estimated by calculating the spatial histogram of the object category for each pixel in the training set. We show examples of estimated $\mathcal{H}$ in our cross-domain traffic scene data sets in Fig. 2.

To take into account smoothness, we define penalty term $\psi(\mathcal{L}(I^{(b)}, p), \mathcal{L}(I^{(b)}, q))$ for assigning labels $\mathcal{L}(I^{(b)}, p)$ and $\mathcal{L}(I^{(b)}, q)$ to two adjacent pixels below:

[1] A set of displacements $w(p)$ defined on the whole image.
[2] A set of SIFT descriptors is defined on the whole image.
[3] A set of annotated labels is defined on the whole image indicating the category of object at pixel p.

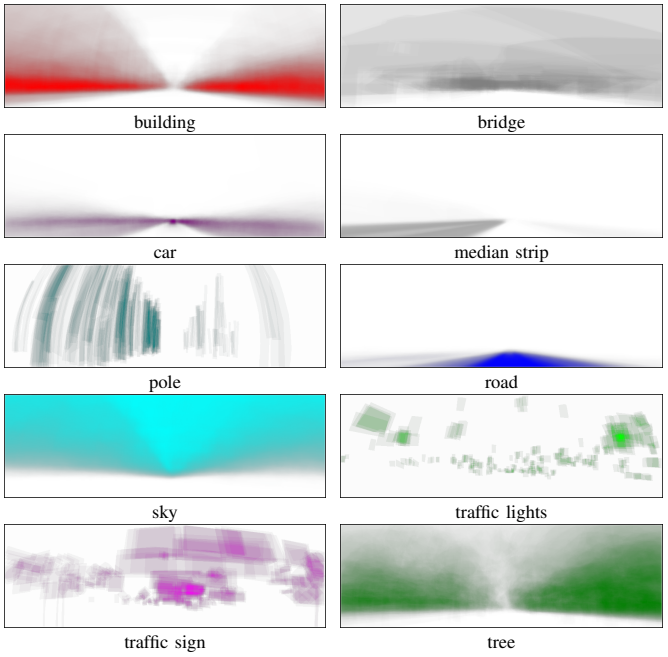


Fig. 2. The statistics for spatial priors of some object categories in our cross-domain traffic scene data set. Note that white means the probability of appearance for this category is zero. The denser the color, the higher the probability.

Fig. 3. Example images from our cross-domain traffic scene data set (first route). Images for two different locations are shown. Each location contains traffic scene images varying from weather and illumination (top to bottom).

- If $\mathcal{L}(I^{(b)},p) \neq \mathcal{L}(I^{(b)},q)$, then

$$\psi(\mathcal{L}(I^{(b)},p),\mathcal{L}(I^{(b)},q)) = e^{-\gamma\|I^{(b)}(p)-I^{(b)}(q)\|_2^2}, \quad (14)$$

  where γ is an image-dependent contrast constant[4].
- If $\mathcal{L}(I^{(b)},p) = \mathcal{L}(I^{(b)},q)$, then

$$\psi(\mathcal{L}(I^{(b)},p),\mathcal{L}(I^{(b)},q)) = 0. \quad (15)$$

To accurately infer the annotaion field $\mathcal{L}(I^{(b)})$, similar to [44], we build a probabilistic MRF model by integrating dense correspondences, spatial prior information and spatial smoothness as follows:

$$\begin{aligned}\min_{\mathcal{L}(I^{(b)})} &\sum_p \phi(\mathcal{L}(I^{(b)},p)) + \alpha\sum_p \theta(\mathcal{L}(I^{(b)},p)) \\ &+\beta\sum_{(p,q)\in\mathcal{E}} \psi(\mathcal{L}(I^{(b)},p),\mathcal{L}(I^{(b)},q)).\end{aligned} \quad (16)$$

Finally, we solve for $\mathcal{L}(I^{(b)})$ by using the belief propagation algorithm.

Remark. Our task of traffic scene understanding involves semantic labeling or segmentation only. In general, traffic scene understanding should also infer the spatial relationship of the recognized objects. This may be a subject of future research.

IV. Experiments

To validate effectiveness of our proposed approach, we conduct our extensive experimental evaluation.

[4]The γ ensures that the exponential term in (14) switches properly between high and low contrasts [49]. Usually, $\gamma = \frac{0.5}{\mathbb{E}[\|I^{(b)}(p)-I^{(b)}(q)\|^2]}$, $\mathbb{E}[\cdot]$ is the expectation taken over image $I^{(b)}$.

A. Data Set and Evaluation Metric

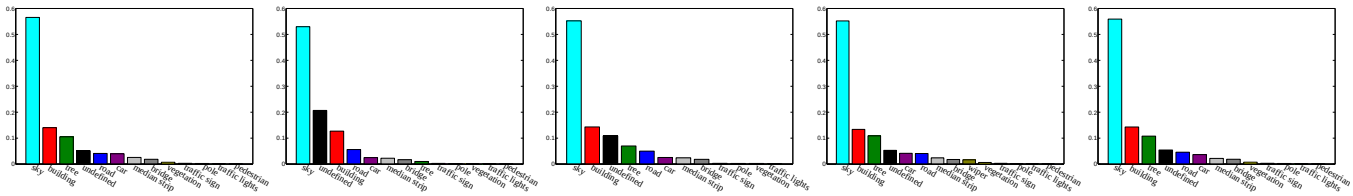
1) Data Set: Our cross-domain traffic scene data set consists of traffic scene images collected from two road routes. The images of the first road route are from five video sequences captured by our test vehicle. It consists of 1,130 traffic scene images of 226 different locations. Both traffic scenes for city and highway are included in this road route. All videos were captured on the same road route, but with different weather and illumination conditions. At each location, we captured images of 5 different conditions, i.e. "sunny day", "night", "snowy day", "rainy night" and "cloudy day", as illustrated in Fig. 3. All of the images are of 856×270 pixels.

The images of the second road route are from two video sequences collected from YouTube. Specifically, the image data consists of 698 traffic scene images of 349 different locations. At each location, 2 different conditions were captured, i.e. "sunny day" and "foggy day", as illustrated in Fig. 4. All of the images are of 640×360 pixels.

In addition, all of the 1,130 images collected from the first route, and 100 images (50 pair-wise images) collected from the second route are manually annotated with LabelMe [50]. There are 13 object categories for the annotated images. Any other object categories are classified as undefined category. The statistics of the annotated object categories are shown in Fig. 5 and Fig. 6. This data set is available online for free to use for research purpose. [5]

2) Evaluation Metrics: To evaluate different deep representations and cross-domain approaches, we calculate Cumulative Matching Characteristic (CMC) curve, which is commonly used as a measure of identification system [41], [51]. Our

[5]www.dabi.temple.edu/~hbling/data/scene-itsc16/benchmark_itsc16.html.

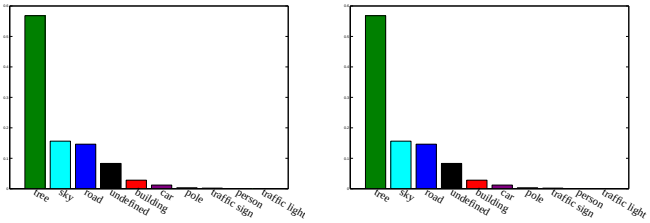


Sunny day Night Rainy night Snowy day Cloudy day

Fig. 6. Statistics for the annotation results of our proposed traffic scene data set (first route) with 13 object categories (sky, building, tree, car, road, median strip, bridge, wiper, vegetation, traffic sign, pole, traffic lights and pedestrian). Any other things are annotated as undefined.

sunny day foggy day

Fig. 4. Example images of foggy day in our cross-domain traffic scene data set (second route).



Sunny day Foggy day

Fig. 5. Statistics for the annotation results of our cross-domain traffic scene data set (second route).

experimental protocol to prepare the CMC curves for each pair-wise domain is to randomly divide the data for each domain (half for training and the other half for testing), with the average performance computed over 10 random splits.

For evaluating scene understanding performance, we use the average per-pixel and per-class recognition rates, which are commonly used as a measure of accuracy of scene understanding systems [44], [52], [53], [54]. The average per-pixel recognition rate $\overline{r}$, which is similar to precision, is computed as

$$\overline{r} = \frac{\sum_I \sum_{p\in\Lambda_I} \mathbb{I}(u(p) = a(p))}{\sum_I \sum_{p\in\Lambda_I} \mathbb{I}(a(p))}, \tag{17}$$

where $a(p)$ is the ground-truth for defined pixel p in image I (some pixels annotated as undefined, as illustrated in Fig. 6), $u(p)$ is the understanding result for pixel p and Λ_I is the lattice of image I. The average per-class recognition rate $\overline{r}_c$ is computed as

$$\overline{r}_c = \frac{\sum_I \sum_{p\in\Lambda_I} \mathbb{I}(u(p) = a(p), a(p) = c)}{\sum_I \sum_{p\in\Lambda_I} \mathbb{I}(a(p) = c)}, \tag{18}$$

where $c \in \{1, \ldots, N\}$.

3) Parameter Settings: For domain adaptation, the dimensionalities p, q, and k are 113, 112, and 100, respectively. In the MRF model in (16), we set $\alpha = 0.10$ and $\beta = 20$ for our experiments.

B. Evaluation on State-of-the-art Pre-trained CNN Models

We select state-of-the-art CNN architectures which are pre-trained on ImageNet (ILSVRC) and Places. For networks pre-trained on ImageNet, we select the top systems in the ILSVRC competitions from 2012 to 2014.

Notice that, to conduct the performance comparison of using all pre-trained networks in the same framework i.e. Caffe [55], we use the pre-trained network of VGG-M [28], which is very similar to the network proposed in [27], to replace the network in [27] and VGG-S [28], which is related to the accurate network from the OverFeat package [56], to replace the network in [56]. The pre-trained networks of VGG-M and VGG-S obtained by Caffe framework are available to download directly.

We use the pre-trained CNN to extract deep representations for the traffic scene images and compare performances of CNNs with different architectures, different layers, and different data sets used to pre-train.

1) Evaluation on Deep Features from Different Layers: We compare different deep representations of our traffic scene images extracted from three different layers: the last convolutional layer (after pooling), the first fully-connected layer (fc6 layer) and the second fully-connected layer (fc7 layer) of AlexNet and VGG-VD-16 pre-trained on ILSVRC-2012. For the AlexNet, we first reshape the $6 \times 6 \times 256$ maps from the output of the last convolutional layer as the 9,216 dimensional vector, and $7\times7\times512$ maps is reshaped as 25,088 dimensional vector for VGG-VD-16. For the two networks, 4,096 dimensional vector is obtained from the two fully-connected layer. All of the vectors are L2-normalized, and these vectors are chosen as the feature vectors. The recognition results (CMC curves) based on these feature vectors for all of the pair-wise domains are shown in Fig. 7. As can be seen in Fig. 7, for the two networks, features of the last convolutional layer get the best performance followed by the fc6 layer and the fc7 layer. The architectures of VGG-M and VGG-S are similar to AlexNet. We use AlexNet and VGG-VD-16 to compare the feature discrimination of different layers.

2) Evaluation on Different Pre-trained CNNs: To compare different pre-trained CNNs, we extract deep representations from the last convolutional layer in the AlexNet, VGG-M, VGG-S and VGG-VD-16 which are pre-trained on ILSVRC-2012. As shown in Fig. 8, the AlexNet, VGG-M and VGG-S all have good performance on various pair-wise domains. To our surprise, the VGG-VD-16 has the worst performance. Karen Simonyan et al [24] have shown that the very deep features have a good generalization when transferred to PASCAL

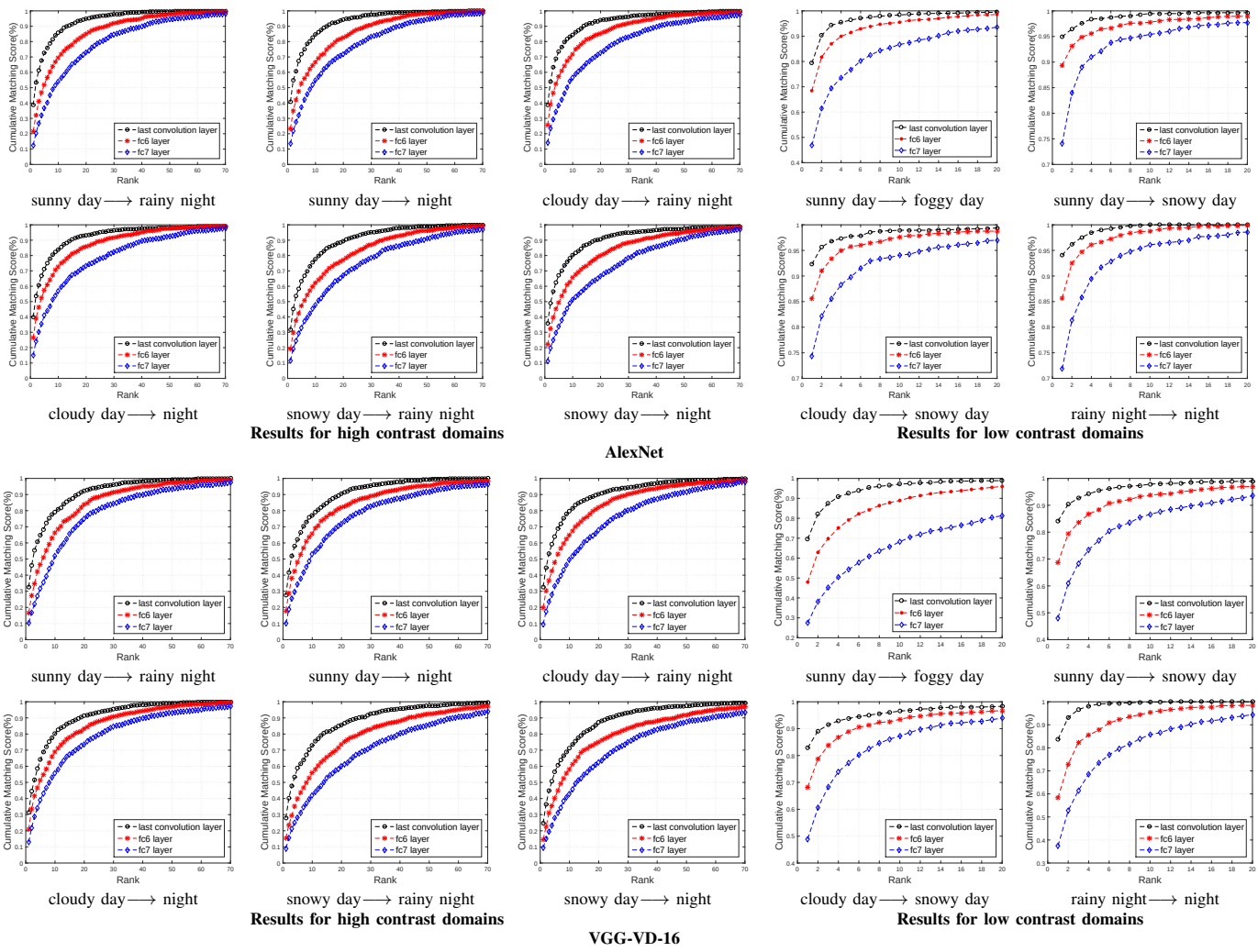


Fig. 7. Recognition results with different layers. For the two networks i.e. AlexNet [22] and VGG-VD-16 [24] experimented in various pair-wise domains of our traffic scene dataset, features of the last convolutional layer achieve the best results followed by the fc6 layer and then the fc7 layer.

VOC-2007 and VOC-2012 benchmarks [57], and image classification benchmarks of Caltech-101 [58] and Caltech-256 [59]. Therefore, we should look at what kinds of pretraining are useful for what tasks.

3) Evaluation on CNNs which are Pre-trained on Different Data Sets: To compare the performance of deep representations extracted from CNNs pre-trained on different data sets, we use the last convolutional layer of the AlexNet which are pre-trained on ILSVRC-2012 [30], Places [23] and Places2 [31]. We also test the deep representations extracted from AlexNet pre-trained on the data set combining Places with ILSVRC-2012 released by MIT Places team. As can be seen in Fig. 8, for high contrast domains, the dataset combining Places with ILSVRC-2012 (1,183 categories) has the best performance. In contrast, there are no significant differences among different large data sets for the low contrast domains.

C. Evaluation on Effects of Fine-tuning CNN

We fine-tune AlexNet pre-trained on a combined data sets of ILSVRC-2012 and Places on our traffic scene data set by using the Caffe framework. We predict 226 or 349 classes for the traffic scene data sets instead of 1,183 for the pre-trained data set. We train the last layer only initialized from random weights. To avoid over-fitting, we use data augmentation as mentioned in Section III-A. We set the initial learning rate as 0.0001 decreasing it by an order of magnitude every 10,000 iterations. We choose the model of 40,000 iterations. We show the performance before and after fine-tuning in Table II. We select the Rank-1, Rank-5 and Rank-10 for comparison. The precision is improved after fine-tuning, particularly the high contrast pair-wise domains are improved significantly, e.g. 9.4 percent improvement for Rank-1 of the cloudy day⟶ rainy night.

D. Comparison Deep Representation and Domain Adaptation with State-of-the-Art

1) Comparison Deep Representations With/without Domain Adaptation: Our traffic scene images may undergo very large appearance variations. The domain-invariant transformation is learned by using the labeled training data from two domains.

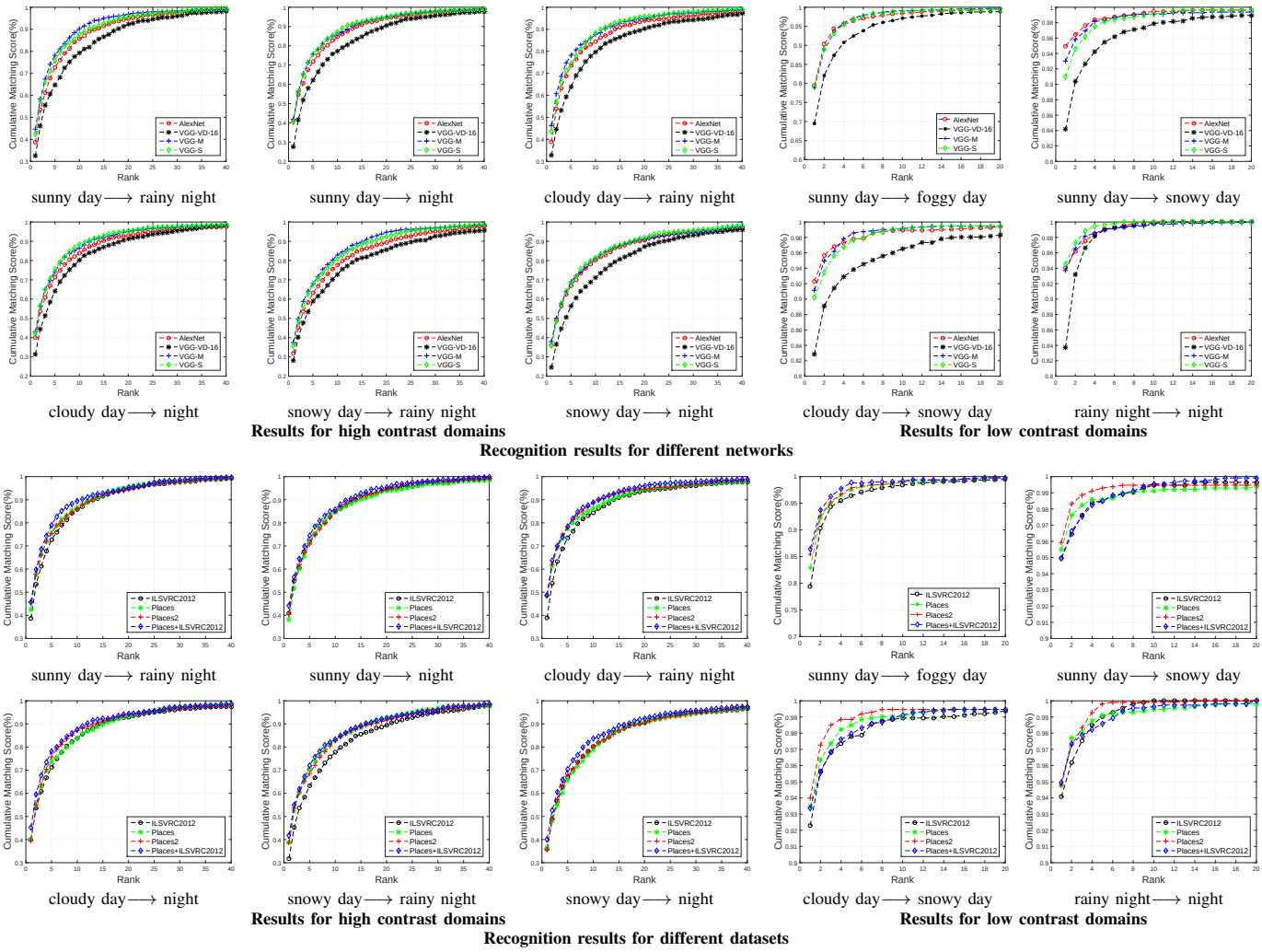


Fig. 8. Recognition results with different networks/datasets. For different networks, the AlexNet, VGG-M [28] and VGG-S [28] have good performance while the VGG-VD-16 has the worst performance. As for different datasets, the dataset combined Places [23] with ILSVRC-2012 [30] obtains the best result for high contrast domains, which contains the most categories i.e. 1,183 categories.

TABLE II
PERFORMANCE BEFORE/AFTER FINE-TUNING(%).

Pair-wise domains	Before			After		
	Rank 1	Rank 5	Rank 10	Rank 1	Rank 5	Rank 10
sunny day⟶ night	44.2	74.6	85.8	47.3	77.6	88.8
sunny day⟶ rainy night	45.7	79.1	89.6	52.5	82.7	90.2
snowy day⟶ night	40.2	70.4	83.6	46.5	73.4	84.3
snowy day⟶ rainy night	41.8	72.0	83.4	51.2	75.1	84.6
cloudy day⟶ night	45.3	78.1	87.7	48.0	81.0	89.6
cloudy day⟶ rainy night	48.8	78.4	88.9	58.2	83.5	91.4
sunny day⟶ snowy day	95.0	98.5	99.6	97.0	98.8	99.6
sunny day⟶ foggy day	86.4	98.9	99.2	88.6	99.3	99.5
cloudy day⟶ snowy day	93.4	98.0	99.1	93.8	98.0	99.3
rainy night⟶ night	95.0	98.6	99.7	96.7	99.4	100

The performance of deep representations before/after cross-domain transformation is compared, as illustrated in Fig. 9. As can be seen in Fig. 9, the performance of deep representations is improved substantially after utilizing the cross-domain transformation, e.g. 12.57 percent improvement for Rank-1. Only conditions of high contrast domains are considered, which are the most challenging.

2) Comparison with Local Invariant Features: Local invariant features have been applied to represent images for matching across appearance changes, e.g. viewpoint and scale. The densely sampled descriptor with compact VLAD encoding is proposed in [17], which has better performance compared with repeatable detection of local invariant features for recognizing the same scene across large appearance changes,

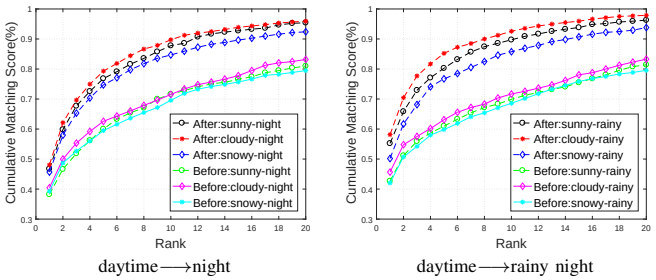


Fig. 9. Comparison deep representations before/after cross-domain transformation.

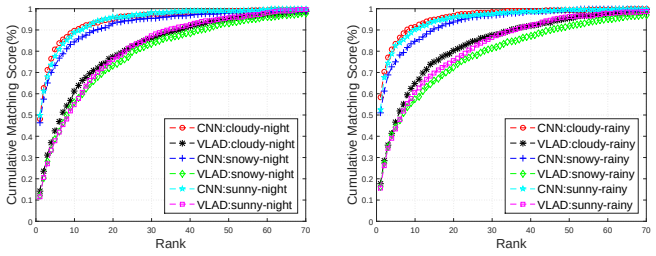


Fig. 10. Comparison with local invariant features. For different challenging scenarios, i.e. the high contrast domains shown in Table 1.

e.g. illumination (day and night times). We compare our deep representations extracted from fine-tuned CNN with this local invariant features. The dense VLAD descriptors of the traffic scene images are computed according to [17].

We compute the dense VLAD descriptors on the original images, rather than after resizing each image to maximum dimension as in [17]. The visual vocabulary of 128 visual words is built from descriptors randomly sampled from our traffic scene dataset using k-means clustering. We fine-tune the CNN using images from our traffic scene data sets. It is helpful to compare the dense VLAD with CNN descriptors on the same level. Unlike [17], the dense VLAD descriptors are not compressed by using PCA because our method uses a way of dimension reduction more robust than PCA. We use the same transformation for comparing different image representations.

Fig. 10 shows the recognition results of different image representations. As can be seen in Fig. 10, the performance of CNN descriptors is better than the dense VLAD descriptors, e.g. 35.67 percent for Rank-1 improvement by using CNN descriptors. These results demonstrate that our method is effective for scene recognition in challenging conditions.

3) Comparison with Subspace Based Transformation Learning Methods: For subspace based transformation learning, we compare our method with two state-of-the-art methods i.e. Geodesic Flow Kernel (GFK) [19] and Subspace Alignment (SA) [18] on our traffic scene data sets. For GFK, the intermediate subspaces are learned along the geodesic direction from one domain to another domain. As for SA, the transformation is learned between subspaces of two domains. For all of the transformation learning methods, the deep representations extracted from the fine-tuned network are used as the input image representations.

As can be seen in Fig. 11, our method has the best performance compared to the state-of-the-art methods. Our method outperforms SA and GFK methods in the high contrast domains. As for the low contrast domains, our method performs slightly better than them. Our method has better performance comparing with the state-of-the-art subspace based methods, particularly for traffic scene images undergoing significant appearance variations.

E. Scene Understanding Results

In this subsection, we conduct the quantitative and qualitative traffic scene understanding experiments. Specifically, for ten different pair-wise domains shown in Table I, half data from each domain are used for learning the cross-domain transformation. Then, the label transfer method is verified on the other half data from each domain. We retrieve $\kappa = 1$ image for each test image in the target domain. The average per-class and per-pixel rates[6] for each condition are shown in Table III.

The major challenge for traffic scene understanding is non-uniform statistics of object categories in a traffic scene. "Texture" classes, such as road, sky, tree etc., constitute the majority of the image pixels, which have no consistent shape but consistent texture. In contrast, "object" classes which are characterized by overall shape occupy a small percentage of the image pixels, e.g. traffic signs, traffic lights and poles. As shown in Table III, for the low contrast domains both object classes ("Bridge", "Median Strip", "Car", "Traffic Sign", "Traffic Lights" and "Pole") and texture categories ("Building", "Road", "Tree", "Sky" and "Vegetation") have good performance. However, for the high contrast domains the performance is decreased especially for the object classes. The recognition rates of poles and traffic signs are significantly different between high and low contrast domains. The main reason is that there are drastic illumination changes for high contrast domains. We show qualitative results in Fig. 12 and 13.

F. Running Time and Implementation Environment

In Table IV we show the running time of each part and the implementation environment/programming language of the proposed algorithm. The fine-tuning is conducted on NVIDIA EVGA GeForce GTX TITAN X GPU by using Caffe framework. Transformation learning, image retrieval (i.e. finding the matching image in another domain given the test image) and label transfer are tested on Intel Core i7-4770 CPU with 16 GB of RAM in Matlab/C++ implementation. As can shown in Table IV, the fine-tuning and domain transformation are learned off-line, and the pre-learned models are used in real traffic. Currently it takes less than two seconds to understand one test image. Further speedup could be achieved through GPU implementation in future work. The dense correspondence/label transfer is implemented by using the method from [60], greatly speeding up the inference.

[6]Images in the target domain are interpreted by label transfer from the source domain. Hence, it would be meaningful to compute the recognition rates for categories which are jointly "owned" by different weather conditions. For example, as can be seen in Fig. 6, the wiper class is only included in snowy day. The recognition rates of 11 classes owned by all weather conditions are reported in Table III, and the recognition results for pedestrian and wiper are not included.

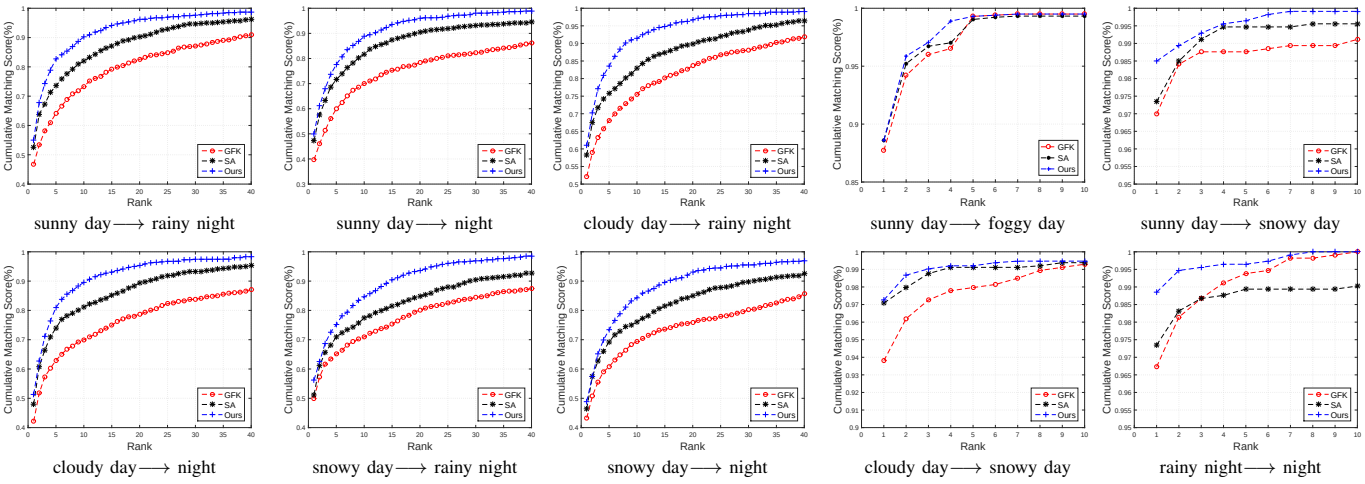


Fig. 11. Comparison with different transformation learning methods. For various challenging scenarios, our method outperforms the state-of-the-art methods i.e. Geodesic Flow Kernel (GFK) [19] and Subspace Alignment (SA) [18].

TABLE III
SCENE UNDERSTANDING RESULTS ON OUR CROSS-DOMAIN TRAFFIC SCENE DATASET (%)

	Bridge	Building	Car	Median Strip	Pole	Road	Sky	Traffic Lights	Traffic Sign	Tree	Vegetation	Per-class	Per-pixel
cloudy day⟶snowy day	97.1	91.6	38.9	68.9	84.0	83.0	98.5	80.6	92.3	88.0	66.3	80.8	90.6
cloudy day⟶rainy night	66.9	71.4	41.7	39.1	39.3	41.9	91.1	33.0	58.8	55.8	15.0	50.4	78.7
cloudy day⟶night	61.8	61.3	34.5	41.6	15.5	43.5	87.5	17.1	63.1	48.9	10.1	44.1	75.7
snowy day⟶night	64.2	71.7	33.8	51.1	16.4	57.0	79.6	13.2	53.1	24.3	9.6	43.1	73.1
snowy day⟶rainy night	80.1	67.7	43.5	51.2	32.2	58.4	81.6	36.5	43.1	38.0	20.8	50.3	72.4
rainy night⟶night	96.8	86.9	42.4	83.2	54.1	81.4	70.9	89.4	92.0	89.2	12.1	72.6	89.5
sunny day⟶night	43.6	69.0	31.3	72.4	23.4	67.6	84.7	10.7	56.5	45.7	27.3	48.4	76.9
sunny day⟶rainy night	42.9	72.9	38.8	68.9	24.2	61.7	80.4	11.0	64.5	48.6	35.8	50.0	72.8
sunny day⟶snowy day	88.9	90.8	36.7	86.8	78.7	83.3	92.9	47.5	94.9	90.6	68.3	78.1	87.7
sunny day⟶foggy day	-	70.1	31.9	-	51.9	86.7	87.1	10.5	46.3	89.0	-	59.2	86.3

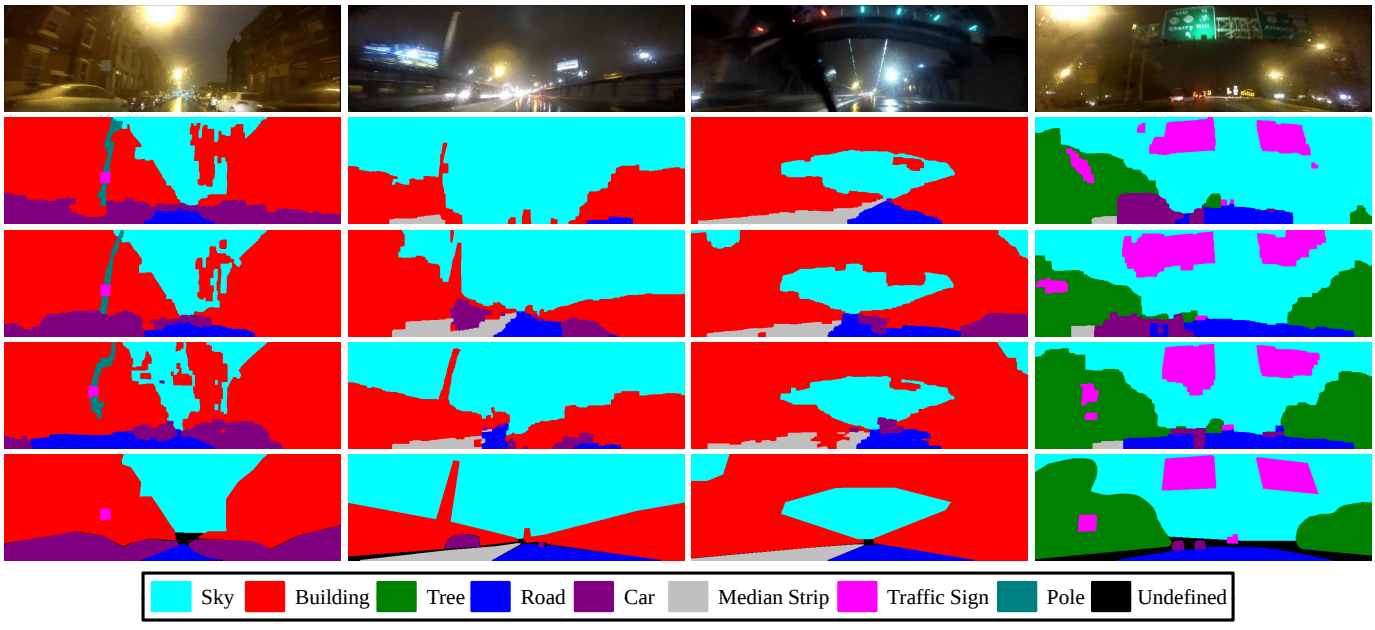


Fig. 12. Some representative scene understanding results of the rainy night scenario. Original images, results for cloudy day⟶ rainy night, results for snowy day⟶ rainy night, results for sunny day⟶ rainy night and human annotation are shown in each row, respectively (top to bottom).

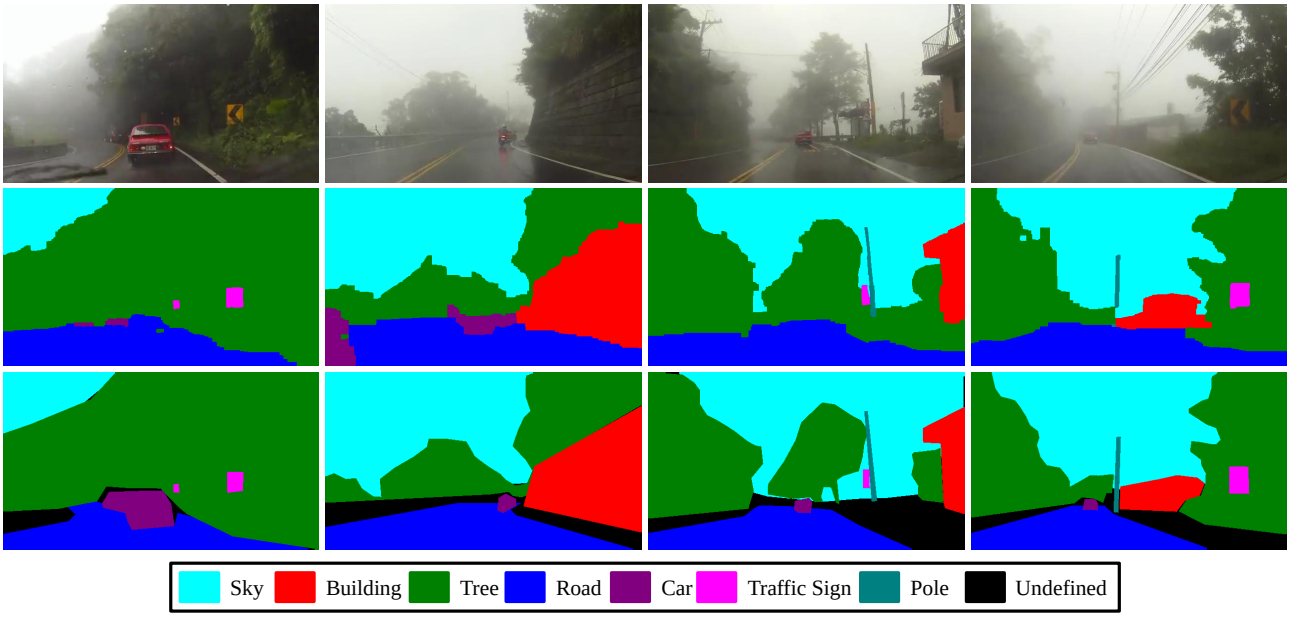


Fig. 13. Some representative scene understanding results of the foggy day scenario. Original images, results for sunny day⟶ foggy day and human annotation are shown in each row, respectively (top to bottom).

TABLE IV
RUNNING TIME (SECOND) AND THE IMPLEMENTATION ENVIRONMENT

	Fine-tune	Transform learning	Image retrieval	Label transfer
Running time	Off-line	Off-line	0.31	1.6
Environment	Caffe	Matlab	Matlab	Matlab/C++

V. CONCLUSION

In this paper we proposed a dense correspondence based transfer learning approach. The approach employs a fine-tuned CNN to extract deep features. It performs cross-domain metric learning and subspace alignment for constructing compact representations to retrieve the cross-domain best matching image. The approach transfers the annotations from the cross-domain best matching image to the test image based on the established dense correspondences between them. We conducted extensive experiments with our new cross-domain traffic scene data set. Experimental results demonstrated the effectiveness of our proposed approach. We hope that our work can pave a new way to traffic scene understanding in challenging driving situations under varying weather and illumination conditions.

The robustness of our proposed approach is dependent upon the dense correspondences based on the SIFT flow. As shown in [61], [62], [63], [64], SCNN [65] is of potential value to learn more powerful representations for finding correspondences. We leave this as our future work.

ACKNOWLEDGMENT

This work was supported in part by the National Natural Science Foundation of China under Grant 61601042, Grant 61402047 and Grant 61528204, and in part by the Beijing Natural Science Foundation under Grant 4162044.

REFERENCES

[1] A. Geiger, C. Wojek, and R. Urtasun, "Joint 3d estimation of objects and scene layout," in *Proc. NIPS*, 2011, pp. 1467–1475.

[2] C. Guo, J. Meguro, Y. Kojima, and T. Naito, "A multimodal adas system for unmarked urban scenarios based on road context understanding," *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 4, pp. 1690–1704, 2015.

[3] J. C. McCall and M. M. Trivedi, "Video-based lane estimation and tracking for driver assistance: Survey, system, and evaluation," *IEEE Trans. Intell. Transp. Syst.*, vol. 7, no. 1, pp. 20–37, 2006.

[4] Y. He, H.Wang, and B. Zhang, "Color-based road detection in urban traffic scenes," *IEEE Trans. Intell. Transp. Syst.*, vol. 5, no. 4, pp. 309–318, 2004.

[5] H. Guan, J. Li, Y. Yu, Z. Ji, and C. Wang, "Using mobile lidar data for rapidly updating road markings," *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 5, pp. 2457–2466, 2015.

[6] R. Marc, G. Dominique, and P. Evangeline, "Generator of road marking textures and associated ground truth applied to the evaluation of road marking detection," in *Proc. ITSC*, 2012, pp. 933–938.

[7] Y. Kang, K. Yamaguchi, T. Naito, and Y. Ninomiya, "Multiband image segmentation and object recognition for understanding road scenes," *IEEE Trans. Intell. Transp. Syst.*, vol. 12, no. 4, pp. 1423–1433, 2011.

[8] Q. Zou, H. Ling, S. Luo, Y. Huang, and M. Tian, "Robust nighttime vehicle detection by tracking and grouping headlights," *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 5, pp. 2838–2849, 2015.

[9] S. Di, H. Zhang, X. Mei, D. Prokhorov, and H. Ling, "A benchmark for cross-weather traffic scene understanding," in *Proc. ITSC*, 2016, pp. 2150–2156.

[10] ——, "Spatial prior for nonparametric road scene parsing," in *Proc. ITSC*, 2015, pp. 1209–1214.

[11] T. Sattler, M. Havlena, F. Radenovic, K. Schindler, and M. Pollefeys, "Hyperpoints and fine vocabularies for large scale location recognition," in *Proc. Int. Conf. Comput. Vis.*, 2015, pp. 2102–2110.

[12] J. Knopp, J. Sivic, and T. Pajdla, "Avoding confusing features in place recognition," in *Proc. Eur. Conf. Comput. Vis.*, 2010, pp. 748–761.

[13] M. Milford, W. J. Scheirer, E. Vig, A. Glover, O. Baumann, J. Mattingley, and D. D. Cox, "Condition-invariant, top-down visual place recognition," in *Proc. IEEE Conf. Robotics and Automation*, 2014, pp. 5571–5577.

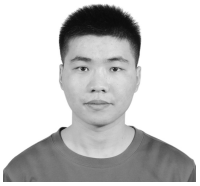
[14] Y. Li, N. Snavely, D. Huttenlocher, and P. Fua, "Worldwide pose estimation using 3d point clouds," in *Proc. Eur. Conf. Comput. Vis.*, 2012, pp. 15–29.

[15] T. Sattler, B. Leibe, and L. Kobbelt, "Improving image-based localization by active correspondence search," in *Proc. Eur. Conf. Comput. Vis.*, 2012, pp. 752–765.

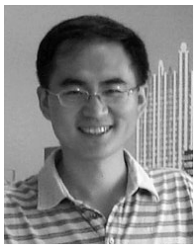
[16] A. R. Zamir and M. Shah, "Accurate image localization based on google maps street view," in *Proc. Eur. Conf. Comput. Vis.*, 2010, pp. 255–268.

[17] A. Torii, R. Arandjelovic, J. Sivic, M. Okutomi, and T. Pajdla, "24/7 place recognition by view synthesis," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 1808–1817.

[18] B. Fernando, A. Habrard, M. Sebban, and T. Tuytelaars, "Unsupervised visual domain adaptation using subspace alignment," in *Proc. Int. Conf. Comput. Vis.*, 2013, pp. 2960–2967.
[19] B. Gong, Y. Shi, F. Sha, and K. Grauman, "Geodesic flow kernel for unsupervised domain adaptation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2012, pp. 2066–2073.
[20] R. Aljundi, R. Emonet, D. Muselet, and M. Sebban, "Landmarks-based kernelized subspace alignment for unsupervised domain adaptation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 56–63.
[21] R. Gopalan, R. Li, and R. Chellappa, "Domain adaptation for object recognition: An unsupervised approach," in *Proc. Int. Conf. Comput. Vis.*, 2011, pp. 999–1006.
[22] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Proc. NIPS*, 2012, pp. 1097–1105.
[23] B. Zhou, A. Khosla, A. Lapedriza, A. Torralba, and A. Oliva, "Learning deep features for scene recognition using places database," in *NIPS*, 2014, pp. 487–495.
[24] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proc. ICLR*, 2015.
[25] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 1–9.
[26] J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, and T. Darrell, "Decaf: A deep convolutional activation feature for generic visual recognition," in *Proc. ICML*, 2014, pp. 647–655.
[27] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 818–833.
[28] K. Chatfield, K. Simonyan, A. Vedaldi, and A. Zisserman, "Return of the devil in the details: Delving deep into convolutional nets," in *arXiv preprint arXiv:1405.3531*, 2014.
[29] A. S. Razavian, H. Azizpour, J. Sullivan, and S. Carlsson, "Cnn features off-the-shelf: an astounding baseline for recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. DeepVision workshop*, 2014, pp. 806–813.
[30] J. Deng, W. Dong, R. Socher, L. Li, K. Li, and F. F. Li, "Imagenet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2009, pp. 248–255.
[31] B. Zhou, A. Khosla, A. Lapedriza, A. Torralba, and A. Oliva, "Places2: A large-scale database for scene understanding," in *http://places2.csail.mit.edu/*, 2015.
[32] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 3431–3440.
[33] H. Noh, S. Hong, and B. Han, "Learning deconvolution network for semantic segmentation," in *Proc. Int. Conf. Comput. Vis.*, 2015, pp. 1520–1528.
[34] S. Zheng, S. Jayasumana, B. R. Paredes, V. Vineet, Z. Su, D. Huang, and P. Torr, "Conditional random fields as recurrent neural networks," in *Proc. Int. Conf. Comput. Vis.*, 2015, pp. 1529–1537.
[35] D. Eigen and R. Fergus, "Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture," in *Proc. Int. Conf. Comput. Vis.*, 2015, pp. 2650–2658.
[36] S. Hong, H. Noh, and B. Han, "Decoupled deep neural network for semi-supervised semantic segmentation," in *Proc. NIPS*, 2015, pp. 1495–1503.
[37] X. Qi, C.-G. Li, G. Zhao, X. Hong, and M. Pietikainen, "Dynamic texture and scene classification by transferring deep image features," *Neurocomputing*, vol. 171, pp. 1230–1241, 2016.
[38] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2014, pp. 580–587.
[39] M. Oquab, L. Bottou, I. Laptev, and J. Sivic, "Learning and transferring mid-level image representations using convolutional neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2014, pp. 1717 – 1724.
[40] H. Abdi, "Partial least square regression, projection on latent structure regression, pls-regression," *Wiley Interdisciplinary Reviews: Computational Statistics*, vol. 2, pp. 97–106, 2010.
[41] S. Liao and S. Z. Li, "Efficient psd constrained asymmetric metric learning for person re-identification," in *Proc. Int. Conf. Comput. Vis.*, 2015, pp. 3685–3693.
[42] B. Fernando, A. Habrard, M. Sebban, and T. Tuytelaars, "Subspace alignment for domain adaptation," in *arXiv preprint arXiv:1409.5241*, 2014.
[43] C. Liu, J. Yuen, and A. Torralba, "Sift flow: Dense correspondence across scenes and its applications," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 5, pp. 978–994, 2011.
[44] ——, "Nonparametric scene parsing via label transfer," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 12, pp. 2368–2382, 2011.
[45] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput.Vis.*, vol. 60, no. 2, pp. 91–110, 2004.
[46] J. Long, N. Zhang, and T. Darrell, "Do convnets learn correspondence?" in *Proc. NIPS*, 2014, pp. 1601–1609.
[47] P. Felzenszwalb and D. P. Huttenlocher, "Efficient belief propagation for early vision," *Int. J. Comput.Vis.*, vol. 70, no. 1, pp. 41–54, 2006.
[48] C. Liu, J. Yuen, and A. Torralba, "Nonparametric scene parsing: label transfer via dense scene alignment," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2009, pp. 1972–1979.
[49] C. Rother, V. Kolmogorov, and A. Blake, "Grabcut-interactive foreground extraction using iterated graph cuts," *ACM Transactions on Graphics*, vol. 23, no. 3, pp. 309–314, 2004.
[50] B. Russell, A. Torralba, K. Murphy, and W. T. Freeman, "Labelme: a database and web-based tool for image annotation," *Int. J. Comput.Vis.*, vol. 77, no. 1, pp. 157–173, 2008.
[51] W. Zheng, X. Li, T. Xiang, S. Liao, J. Lai, and S. Gong, "Partial person re-identification," in *Proc. Int. Conf. Comput. Vis.*, 2015, pp. 4678–4686.
[52] J. Tighe and S. Lazebnik, "Superparsing: Scalable nonparametric image parsing with superpixels," *Int. J. Comput.Vis.*, vol. 101, no. 2, pp. 329–349, 2013.
[53] J. Yang, B. Price, S. Cohen, and M. Yang, "Context driven scene parsing with attention to rare classes," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2014, pp. 3294–3301.
[54] J. Tighe, M. Niethammer, and S. Lazebnik, "Scene parsing with object instances and occlusion ordering," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2014, pp. 3748–3755.
[55] Y. Jia, "Caffe: An open source convolutional architecture for fast feature embedding. http://caffe.berkeleyvision.org/," 2013.
[56] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. LeCun, "Overfeat:integrated recognition, localization and detection using convolutional networks," in *Proc. ICLR*, 2014, p. 16.
[57] M. Everingham, S. Eslami, L. V. Gool, C. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes challenge: a retrospective," *Int. J. Comput.Vis.*, vol. 111, no. 1, pp. 98–136, 2015.
[58] F. F. Li, R. Fergus, and P. Perona, "Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshop of Generative Model Based Vision*, 2004, p. 178.
[59] G. Griffinand, A. Holub, and P. Perona, "Caltech-256 object category dataset," in *Technical Report 7694, California Institute of Technology*, 2007.
[60] J. Kim, C. Liu, F. Sha, and K. Grauman, "Deformable spatial pyramid matching for fast dense correspondences," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2013, pp. 2307–2314.
[61] P. Agrawal, J. Carreira, and J. Malik, "Learning to see by moving," in *Proc. Int. Conf. Comput. Vis.*, 2015, pp. 37–45.
[62] J. Flynn, I. Neulander, J. Philbin, and N. Snavely, "Deepstereo: Learning to predict new views from the world's imagery," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 5515–5524.
[63] D. Jayaraman and K. Grauman, "Learning image representations tied to ego-motion," in *Proc. Int. Conf. Comput. Vis.*, 2015, pp. 1413–1421.
[64] J. Zbontar and Y. LeCun, "Stereo matching by training a convolutional neural network to compare image patches," *Journal of Machine Learning Research*, vol. 17, no. 1, pp. 1–32, 2016.
[65] S. Chopra, R. Hadsell, and Y. LeCun, "Learning a similarity metric discriminatively, with application to face verification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2005, pp. 539–546.



Shuai Di is a PhD student jointly training in the Beijing University of Posts and Telecommunications, and the Department of Computer and Information Sciences, Temple University. His research interests include computer vision and pattern recognition and their applications in intelligent vehicles.



Honggang Zhang (SM'12) received the B.S. degree from the Department of Electrical Engineering, Shandong University, in 1996, the masters and Ph.D. degrees from the School of Information Engineering, Beijing University of Posts and Telecommunications (BUPT), in 1999 and 2003, respectively. He was a Visiting Scholar with the School of Computer Science, Carnegie Mellon University, from 2007 to 2008. He is currently an Associate Professor and the Director of Web Search Center with BUPT. He published more than 30 papers on TPAMI, SCIENCE, *Machine Vision and Applications*, AAAI, ICPR, ICIP. His research interests include image retrieval, computer vision, and pattern recognition.

Chun-Guang Li received the B.E. degree in telecommunication engineering from the Jilin University in 2002 and the Ph.D. degree in signal processing from the Beijing University of Posts and Telecommunications (BUPT) in 2007. Currently, he is an associate professor with the School of Information and Communication Engineering, BUPT. From July 2011 to April 2012, he visited the Visual Computing group, Microsoft Research Asia. From December 2012 to November 2013, he visited the Vision, Dynamics, and Learning lab, the Johns Hopkins University. His research interests are statistical signal processing and machine learning. He is a member of the IEEE, ACM, and CCF.

Xue Mei (SM'14) received the B.S. degree in electrical engineering from the University of Science and Technology of China, Hefei, China, and the Ph.D. degree in electrical engineering from the University of Maryland, College Park, MD, USA. He was with the Automation Path-Finding Group in Assembly and Test Technology Development and the Visual Computing Group, Intel Corporation, USA, from 2008 to 2012. He is currently a Senior Research Scientist with the Future Mobility Research Department, Toyota Research Institute, Ann Arbor, MI, USA, a Toyota Technical Center division. He serves as an Adjunct Professor with Anhui University, Hefei. His current research interests include computer vision, machine learning, and robotics with a focus on intelligent vehicles research. Dr. Mei was an Area Chair of the Winter Conference on Computer Vision in 2015 and 2016, and a Lead Organizer of the My Car Has Eyes: Intelligent Vehicle With Vision Technology Workshop at the Asian Conference on Computer Vision in 2014. He serves as a Lead Guest Editor of the Special Issue on Visual Tracking of *Computer Vision and Image Understanding*.

Danil Prokhorov (SM'02) was a Research Engineer with the St. Petersburg Institute for Informatics and Automation, Russian Academy of Sciences, Saint Petersburg, Russia. He has been involved in automotive research since 1995. He was an Intern with the Scientific Research Laboratory, Ford Motor Company, Dearborn, MI, USA, in 1995. In 1997, he became a Research Staff Member with Ford Motor Company, where he was involved in application-driven research on neural networks and other machine learning methods. Since 2005, he has been with the Toyota Technical Center, Ann Arbor, MI, USA. He is currently in charge of the Department of Future Mobility Research, Toyota Research Institute, Ann Arbor. He has authored over 100 papers in various journals and conference proceedings and holds many patents in a variety of areas. Dr. Prokhorov served as the International Neural Network Society President from 2013 to 2014, and was a member of the IEEE Intelligent Transportation Systems Society Board of Governors, a U.S. National Science Foundation Expert, and an Associate Editor/Program Committee Member of many international journals and conferences.

Haibin Ling received the B.S. degree in mathematics and the M.S. degree in computer science from Peking University, China, in 1997 and 2000, respectively, and the Ph.D. degree in computer science from the University of Maryland, College Park, in 2006. From 2000 to 2001, he was an Assistant Researcher with Microsoft Research Asia. From 2006 to 2007, he was a Post-Doctoral Scientist with the University of California at Los Angeles. After that, he joined Siemens Corporate Research as a Research Scientist. Since Fall 2008, he has been with Temple University, where he is currently an Associate Professor. His research interests include computer vision, medical image analysis, human-computer interaction, and machine learning. He received the Best Student Paper Award at the ACM Symposium on User Interface Software and Technology in 2003, and the NSF CAREER Award in 2014. He has served on the editorial board of *IEEE Trans. on Pattern Analysis and Machine Intelligence* and *Pattern Recognition*, and Area Chairs for CVPR 2014 and CVPR 2016.