Exemplar Network: A Generalized Mixture Model

Chikao Tsuchiya
Nissan Motor Co., Ltd.
CSAIL, MIT
Cambridge, USA 02139

Tomasz Malisiewicz,
and Antonio Torralba
CSAIL, MIT
Cambridge, USA 02139

***Abstract*—We present a non-linear object detector called Exemplar Network. Our model efficiently encodes the space of all possible mixture models, and offers a framework that generalizes recent exemplar-based object detection with monolithic detectors. We evaluate our method on the traffic scene dataset that we collected using onboard cameras, and demonstrate an orientation estimation. Our model has both the interpretability and accessibility necessary for industrial applications. One can easily apply our method to a variety of applications.**

I. Introduction

The interpretability and usability of algorithms are important, especially, in the case of industrial applications. State-of-the-art object detection frameworks such as Deformable Part Models (DPMs)[3] are not necessarily interpretable and usable for practitioners. Contrarily, exemplar-based approaches, such as ensemble of Exemplar-SVMs (ESVM)[7] and ensemble of LDA (ELDA)[6], seem more plausible, not only for researchers, but also practitioners. In this paper, we focus on exemplar-based approaches for the sake of better industrial applicability.

ESVM and ELDA learn a classifier for each exemplar, which enables the exemplar-based detectors to deal with intra-class variation better than monolithic detectors[1]. Therefore their detection ability is on par with DPMs[6].

From the viewpoint of mixture models, exemplar-based methods can be seen as an extreme. Traditional mixture modeling breaks up N training samples into K disjoint groups (or "mixture components") and learns a separate model for each group. In that sense, exemplar-based methods are the case where $K = N$. What if instead of committing to a rigid set of mixture components, we instead stored all possible mixture components which could be generated by N positive examples? The naive approach would require enumerating all possible 2^N mixture components. In this paper, we present a model that efficiently performs a search over "the space of all mixture components" without the need to explicitly store an exponentially large number of mixture parameters. Because our model is able to handle this large space of models by only requiring the storage of individual exemplars, we call it the *Exemplar Network* (EN). Our model finds the best possible mixture for each query point at test time.

Meanwhile, exemplar-based methods have another big advantage over traditional methods. That is "meta-data transfer" which provides an easy but efficient way to estimate the attribute of an object. Of course, EN can take advantage of meta-data transfer in more sophisticated way.

In the remainder of the paper, we describe related work in Section 2, and introduce preliminaries for our approach in Section 3. In Section 4, we introduce the concept of Exemplar Network, which is evaluated in Section 5.

II. Related Work

Since the introduction of the Histogram of Oriented Gradients (HOG)[1], the computer vision community has explored a wide range of different models for visual object detection: Monolithic detectors[1], Mixture models[3], [5], Ensemble of Exemplar-SVMs[7](see Fig.1). Both Mixture models and Monolithic models (which are a special case of mixtures with 1 component), are rigid model structures: the representational power of the detector (i.e. the number of mixtures) must be specified before training. To make things worse, setting such parameters requires knowledge about the underlying appearance variations in the data.

Collecting big data is hard work; annotating images requires an immense amount of time and effort. Some researchers have tried collecting an order of magnitude more positive data, but their results indicate that just throwing "big data" at current frameworks does not give a significant boost in performance[11]. Furthermore, although useful meta-data accompanies big data in many cases, very few researchers made use of them[7], [6], [**?**].

Exemplar-based methods such as ESVM and ELDA are flexible because the representational power grows with the addition of new training data. Furthermore, these kinds of methods enable attribute estimation via meta-data transfer. Hariharan et al.[6] showed an example of appearance transfer that replaces the detected objects with the most similar exemplars. Malisiewicz et al.[7] demonstrated segmentation estimation and 3D model transfer. Tighe et al.[**?**] proposed an image parsing using a combination of region-level features and segmentation transfer based on exemplar-based object detectors. Although they used only one meta-data associated with the exemplar that is the most similar to a query point, we believe that summarizing multiple meta-data enables more accurate attribute estimation.

Inspired by the success of exemplar-based methods, we present the more general machinery of Exemplar Network. Exemplar Network's complexity automatically grows with the data without the need to make any strong category-specific or object-specific modeling assumptions beforehand. In order to make training more efficient than in ESVM, we use the recent observation that Linear Discriminant Analysis (LDA) can give results which are comparable to SVMs, but at a significant training time discount[4], [6]. Our model bears

1051-4651/14 $31.00 © 2014 IEEE
DOI 10.1109/ICPR.2014.113

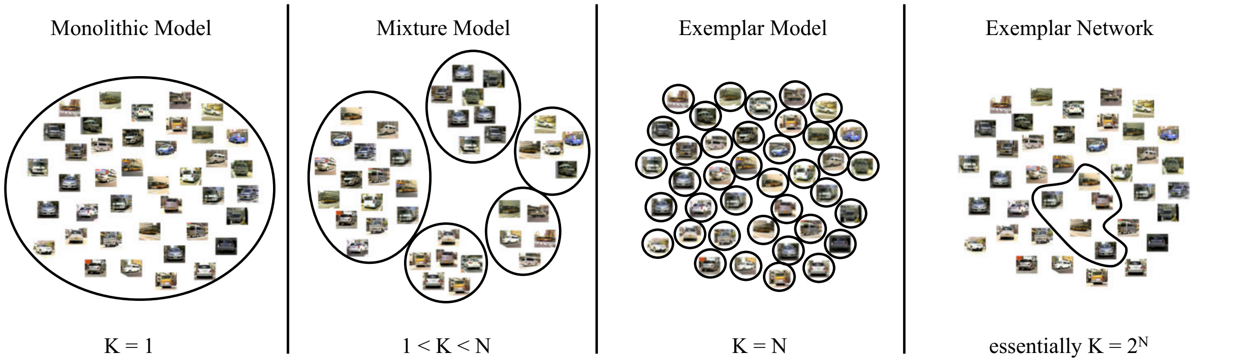

Fig. 1. A spectrum of models from the most rigid to the most flexible. Mixture models provide a unified view for both monolithic models and exemplar models. However, they require specifying the number of mixtures beforehand. In contrast, our model, the Exemplar Network, doesn't require specifying the representational power, where the optimal component is automatically determined at test time.

strong resemblance to Exemplar Theory from human categorization research, which states that subjects store individual category exemplars in memory, and classification is based on the similarity of inputs to stored exemplars[8], [9].

III. Preliminaries

A. Notations

Let $\mathbf{X} = [\mathbf{x}_1 \mathbf{x}_2 \cdots \mathbf{x}_N] \in \mathcal{R}^{N \times F}$ be a training set for N data points and F dimensional features, and $y_i = \{-1, 1\}$ be a class label of $\mathbf{x}_i$. In our notation, $\mathbf{x}_i$ is a training example, and $\mathbf{x}$ without the subscript is the query point that we wish to classify. Note that we write the vector of all ones as $\mathbf{1}_N$, the vectors of all zeros as $\mathbf{0}_N$, and the i-th Euclidean basis vector as $\mathbf{e}_i$.

Our models will operate in whitened space. We map all data points from their original anisotropic feature space to an isotropic 0-mean and unit variance space using the whitening transform: $\hat{\mathbf{x}}_i = \mathbf{\Sigma}^{-\frac{1}{2}}(\mathbf{x}_i - \boldsymbol{\mu})$. The whitening transform requires a covariance matrix $\mathbf{\Sigma} \in \mathcal{R}^{F \times F}$ and a mean vector $\boldsymbol{\mu} \in \mathcal{R}^F$, which can be learned from a large and generic dataset of images[4], [5]. For convenience, we write the whitened training data as $\hat{\mathbf{X}} = [\hat{\mathbf{x}}_1 \cdots \hat{\mathbf{x}}_N]$, and the whitened query as $\hat{\mathbf{x}}$.

B. LDA

We briefly review LDA since it is crucial to our final algorithm. The LDA model assumes that the data points are drawn from a Gaussian distribution and each class has equal covariances. Fisher showed that under these modeling assumptions, the optimal decision function $F_{LDA}(\cdot)$ is linear:

$$f_{LDA}(\mathbf{x}; \mathbf{w}_{LDA}) = \mathbf{w}_{LDA}^T \mathbf{x} \tag{1}$$

$$\text{where} \quad \mathbf{w}_{LDA} = \mathbf{\Sigma}^{-1}(\boldsymbol{\mu}_+ - \boldsymbol{\mu}) \tag{2}$$

We write $\boldsymbol{\mu}_+$ to be the mean of the positive class, and $\boldsymbol{\mu}$ to be the mean of the negatives.

We note that if we apply the LDA classifier to $\mathbf{x}-\boldsymbol{\mu}$ instead of directly to $\mathbf{x}$, then the score is only shifted by a constant and does not affect the ordering of examples: $f_{LDA}(\mathbf{x}-\boldsymbol{\mu}) = f_{LDA}(\mathbf{x}) + c$ for a constant $c \in \mathcal{R}$. Hence, we can write an equivalent whitened LDA classifier:

$$f_{LDA}(\hat{\mathbf{x}}) = \hat{\boldsymbol{\mu}}_+ \hat{\mathbf{x}} \tag{3}$$

C. ELDA

The representational power of LDA is limited because the class of possible decision boundaries is restricted to linear separators. Hariharan et al.[6] showed that one can obtain an significant increase in performance with an exemplar LDA model by training an ensemble of LDA classifiers:

$$f_{ELDA}(\mathbf{x}; \mathbf{X}) = \max_{i \in \{1,\ldots,N\}} \mathbf{w}_i^T \mathbf{x} \tag{4}$$

$$\text{where} \quad \mathbf{w}_i = \mathbf{\Sigma}^{-1}(\mathbf{x}_i - \boldsymbol{\mu}) \tag{5}$$

Note that ELDA can also written as a whitened classifier:

$$f_{ELDA}(\hat{\mathbf{x}}, \hat{\mathbf{X}}) = \max_{i \in \{1,\ldots,N\}} \hat{\mathbf{x}}_i^T \hat{\mathbf{x}} \tag{6}$$

ELDA can be viewed as a mixture model where there are as many mixtures as there are training examples, $K = N$. In addition, LDA can be viewed as a mixture model with one mixture, $K = 1$.

IV. Exemplar Networks

LDA and ELDA represent two extremes for mixture models. There has also been significant work in learning mixture components for other K. In this section, we present a framework based on LDA that infers mixture components on-the-fly.

A. Model

We can write both LDA and ELDA as the following unified equation:

$$f(\hat{\mathbf{x}}; \alpha) = \sum_{i=1}^{N} \alpha_i \hat{\mathbf{x}}_i^T \hat{\mathbf{x}} \tag{7}$$

where α_i determines the degree that $\mathbf{x}_i$ belongs in the mixture component. LDA and ELDA are the special case of eq.7:

$$f(\hat{\mathbf{x}}; \alpha) = \begin{cases} f_{LDA}(\hat{\mathbf{x}}) & (\alpha = \mathbf{1}_N) \\ f_{ELDA}(\hat{\mathbf{x}}) & (\alpha_i = \mathbf{e}_i) \end{cases} \tag{8}$$

In LDA, all data points contribute to the model equally, while in ELDA every example is independent and there is no sharing.

The Exemplar Network generalizes these ideas to work in a significantly larger space of mixtures. In order to classify a query $\hat{\mathbf{x}}$, we propose to search over all possible mixtures at test time:

$$f_{EN}(\hat{\mathbf{x}}) = c + \sum_{i=1}^{N} max\left(a_i \hat{\mathbf{x}}_i^T \hat{\mathbf{x}} + b_i, 0\right) \tag{9}$$

where $max(\cdot, \cdot)$ is the function that returns the larger value of the two arguments, a_i and b_i are the hyperparameters that adjust the degree that $\hat{\mathbf{x}}_i$ is used in a decision, and c is another hyperparameter that only shift the final score. One can see $a_i \hat{\mathbf{x}}_i^T \hat{\mathbf{x}} + b_i$ as an activity of exemplar $\hat{\mathbf{x}}_i$.

B. *Optimization*

To get the appropriate hyperparameters $\mathbf{a}, \mathbf{b}$ and c in eq.9, we use the following cost function:

$$\begin{aligned} L(\mathbf{a}, \mathbf{b}, c) = & \sum_{i=1}^{N} max(1 - f_{EN}(\hat{\mathbf{x}}_i) y_i, 0) \\ & + \gamma_a \sum_{i=1}^{N} |a_i| + \gamma_b \sum_{i=1}^{N} |b_i| \end{aligned} \tag{10}$$

where the first term is the hinge loss of the classification function (eq.9) and the second and third terms are the regularization terms. In the following experiments, we used stochastic gradient descent to minimize eq.10. Note that this optimization is needed only in the training step, not in the test or execution steps.

C. *Cascade for Speed-up*

Similar to exemplar-based methods such as ESVM, our method takes a little bit more time to evaluate all stored exemplars. To deal with this, we can use a cascade with LDA. Our cascade procedure is as follows: we first apply LDA to all windows during performing sliding window search over an image, then apply EN only to the top W windows on the LDA stage. W is the parameter to adjust how many windows can pass through the LDA stage and can be evaluated by EN. In this sense, EN works as a rescorer.

As we described above, LDA's representational power is limited because LDA is a monolithic detector. This means the cascade with LDA may fail in diverse datasets where the inner-class variance is large, but we found that this scheme works well on our dataset unless W is too small.

D. *Meta-data Transfer*

Meta-data transfer is the most distinctive and useful feature of exemplar-based methods including EN. Although the existing researches[7], [6], [**?**] transferred only one meta-data of the exemplar that is the most similar to a query, we propose transferring multiple meta-data. In the procedure of EN, more than one exemplar similar to the query can be activated with different activities (eq.9). Using these activities, we can simply perform a weighted average over the meta-data to estimate an attribute.

TABLE I. THE NUMBER OF OBJECTS IN THE TRAFFIC SCENE DATASET (ONBOARD). THE PARENTHETIC NUMBERS INDICATE THE NUMBER OF OCCLUDED OBJECTS.

	train	test
car	910 (378)	880 (293)
person	1687 (370)	1561 (340)

TABLE II. THE DISTRIBUTION OF ORIENTATIONS IN THE TRAFFIC SCENE DATASET (ONBOARD).

car	0°	45°	90°	135°	180°	225°	270°	315°	N/A
train	51	51	161	98	51	188	172	137	1
test	61	48	179	96	66	195	148	86	1

person	0°	45°	90°	135°	180°	225°	270°	315°	N/A
train	51	51	161	98	51	188	172	137	1
test	61	48	179	96	66	195	148	86	1

In the following experiments, we performed two different meta-data transfers: bounding box transfer and orientation transfer. As for the bounding box transfer, we estimate the bounding box of a target object by averaging all bounding boxes associated with activated exemplars. This procedure is implicitly applied to all the experiments in this article. As for the orientation transfer, we demonstrate it in Section V-E.

V. EVALUATION

A. *Dataset*

We captured images using cameras that are mounted on a car and made a traffic scene dataset that we call ONBOARD dataset. The dataset contains more than 1,700 cars and 3,000 people (Table I). The resolution of each image is 1024 × 1024. Furthermore, we annotated the orientations of the objects in 8 directions (every 45°), the distribution of which is shown in Table II. The dataset is split into a training set and a test set. Unless otherwise noted, we trained detectors with the training set and tested them with the test set in the following experiments.

B. *Cascade with LDA*

We first evaluate the cascade with LDA and EN by changing the parameter W described in Section IV-C. For the evaluation, a detection is deemed correct only if the overlap ratio between the predicted bounding box and ground truth bounding box exceed 0.5, which is exactly the same as the criteria in PASCAL VOC[2].

The LDA detector was trained using all exemplars included in the EN. We excluded occluded exemplars during the training of EN, because we observed that the EN without occluded exemplars slightly outperformed the EN with occluded ones. This issue needs more investigation in the near future.

Fig.2 and Fig.3 show the number of inner products per image and the average precisions (APs) when changing W from 1,000 to 10,000. APs stay steady even if we change W. Using a smaller W results in less computational cost, but has little effect on APs. This result suggests that we can choose a small W for speed-up without significant accuracy defection. In the rest of experiments, we used $W = 1,000$ for efficiency of evaluation.

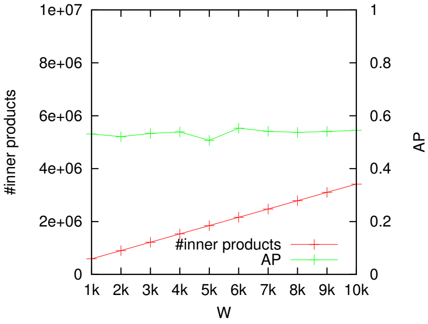

Fig. 2. The number of inner products per image (left axis) and APs (right axis) when changing W on the car detection task.

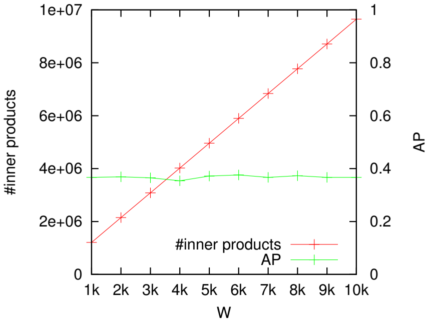

Fig. 3. The number of inner products per image (left axis) and APs (right axis) when changing W on the person detection task.

C. Background Statistics

Similar to LDA, EN requires a covariance matrix $\boldsymbol{\Sigma}$ and mean vector $\boldsymbol{\mu}$ of the background, so their performances obviously depend on those statistics. We are interested in how the deference of background statistics affects the performance of the detectors. To evaluate this, we computed $\boldsymbol{\Sigma}$ and $\boldsymbol{\mu}$ from the PASCAL VOC dataset and our dataset, then configured two different ENs. The APs of those two ENs on the car detection task and the person detection task, are shown in Fig.4 and Fig.5, respectively.

On the car detection task, no significant differences were observed. Surprisingly, on the person detection task, we observed that the EN using the background statistics from the PASCAL VOC dataset slightly outperformed the EN using those from our dataset. We suspect that the limited size of our dataset affected the accuracy of the statistics. This observation suggests that we should use as large a dataset as we can to compute the background statistics, even if it's different from the target dataset. Evaluating EN, after expanding our dataset, is very interesting research issue. We would like this to be our future work. We used the background statistics from the PASCAL VOC dataset for the following experiments.

D. Object Detection

EN and ELDA are different from LDA in that EN and ELDA are exemplar-based methods but LDA is a monolithic detector. Furthermore, EN uses multiple exemplars at a time to score a detection, while ELDA uses only one exemplar. In this section, we compare the detection performance of EN, LDA and ELDA. EN and LDA are configured as they are in the previous experiment. As for ELDA, it holds exactly the same exemplars as EN, but it uses only one exemplar at a time as shown in eq.6.

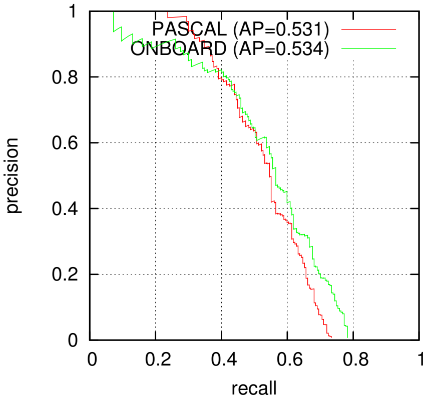

Fig. 4. The difference of background statistics on the car detection task. ONBOARD means our traffic scene dataset. There is no significant difference between PASCAL(AP=0.531) and ONBOARD(AP=0.534).

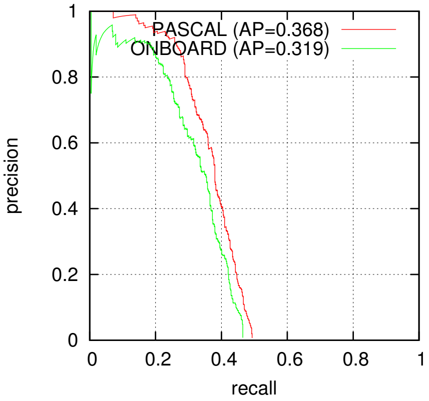

Fig. 5. The difference of background statistics on the person detection task. ONBOARD means our traffic scene dataset. On the contrary to the car detection task, PASCAL(AP=0.368) is slightly better than ONBOARD(AP=0.319).

Fig.6 and Fig.7 show the ROC curves of this experiment. From these results, it's clear that exemplar-based methods outperform monolithic detectors. On the car detection task, EN and ELDA were comparable to each other. Meanwhile, EN marked much better AP than ELDA on the person detection task. From these observations, we can say using multiple exemplars contributes to improving detection performance. On this point, our results and those of Hariharan et al.[6] are consistent, though they trained one LDA detector for each cluster.

To enable readers to compare EN with other methods, we evaluated EN with PASCAL VOC 2007 dataset. The method was evaluated only with person and car categories in accordance with our dataset. The configuration of EN is the same as in the previous experiment except that the number of exemplars was limited to 3,000 to reduce the computational

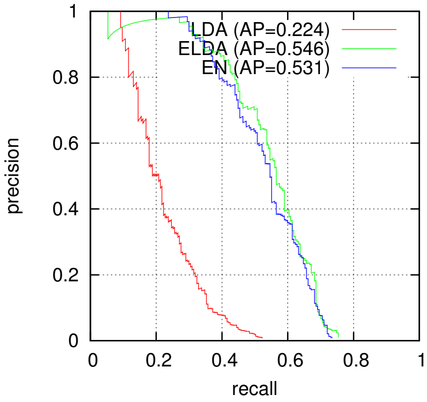

Fig. 6. ONBOARD object detection result for car category.

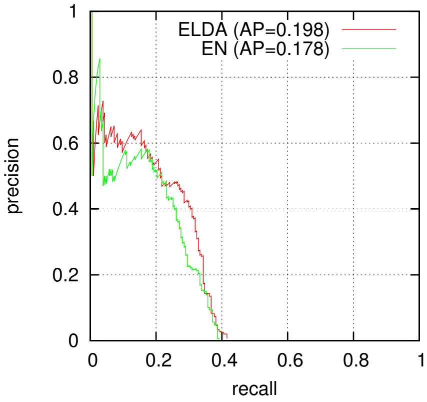

Fig. 8. ONBOARD orientation estimation result for car category.

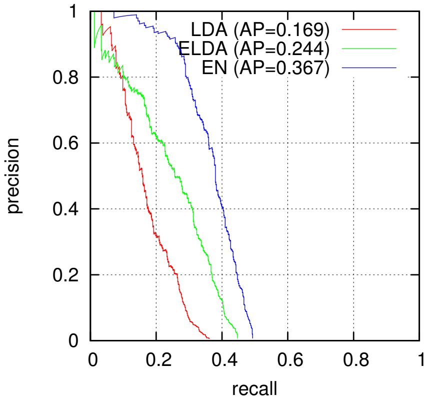

Fig. 7. ONBOARD object detection result for person category.

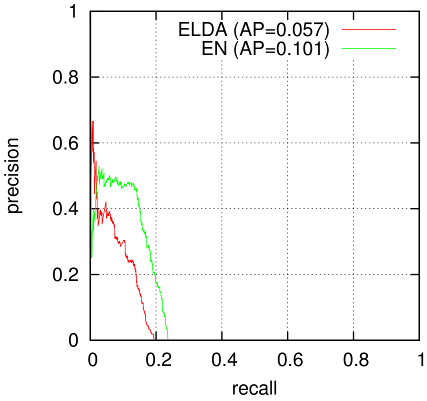

Fig. 9. ONBOARD orientation estimation result for person category.

cost. As shown in Table III, EN was competitive with ESVM in the car category but was slightly below DPM. However, EN didn't work well in the person category. One possible reason is that the cascade with LDA couldn't deal with the diversity of the person category. In fact, LDA's AP of the person category was much worse than the other methods. Of course, its possible to apply EN without the cascade in the same way as ESVM, but it is not practical from the viewpoint of computational cost. This emphasizes the importance of reducing the computational complexity of EN.

E. Orientation Estimation

We demonstrate an orientation estimation based on metadata transfer of EN. As described above, each exemplar has an orientation attribute which ranges from 0° to 315° with 45° intervals. We can simply estimate the orientations of objects by using a weighted voting scheme. In order to evaluate estimations, we scored each estimation as true if it is within the ground truth ±45°. Note that this task implicitly includes the detection task shown in the previous section; therefore this task is much harder than the detection task.

TABLE III. PASCAL VOC 2007 OBJECT DETECTION RESULT.

	person	car
LDA	0.065	0.189
EN	0.097	0.415
ESVM [7]	0.169	0.411
LDPM [3]	0.362	0.502

The results of the orientation estimation are shown in Fig.8 and Fig.9. Compared to the results of the object detection task, the APs of both EN and ELDA dropped. However, considering that the orientation estimation task is much harder than the object detection task, this can be regarded as reasonable. Although EN is slightly inferior to ELDA on the car task, EN outperformed ELDA on the person task. Taking into account that EN obviously outperformed ELDA on the person detection task, we can't conclude that EN is superior to ELDA in terms of orientation estimation. Investigating this issue is an interesting area for future work.

Finally, we demonstrate orientation estimation in Fig.10. The red bounding boxes indicate the detections with the best score, and the yellow ones indicate the other detections. In the bottom of each detection, we show the top 5 activated exemplars for the best detection (drawn in red). All these examples except the one on bottom right, are the successful cases. In the failure case, the orientation of the truck in the left lane was estimated incorrectly. The estimation is based only on the appearance of the object, thus it's hard to estimate the orientations of objects like trucks and buses. On the other hand, the orientations of sedans and hatchbacks are relatively

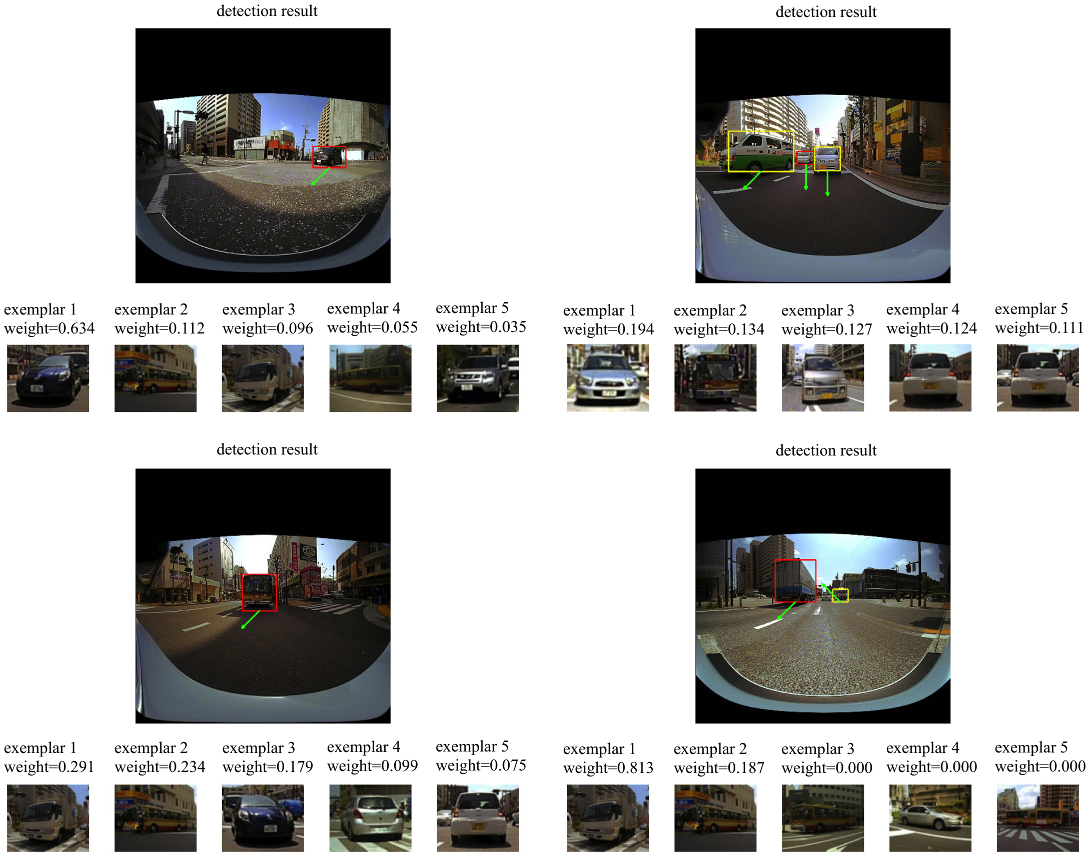

Fig. 10. The examples of orientation estimation on the car detection task. In each example, the red bounding box indicates the top detection, and the yellow ones indicate the other detections. In the bottom of each detection, the top 5 activated exemplars for the top detection are illustrated.

easily estimated.

VI. Conclusion

EN, a novel framework that seamlessly connects LDA and ELDA, was proposed. We evaluated our proposed method with our traffic scene dataset, and demonstrated the orientation estimation based on meta-data transfer. EN is a general and fairly simple framework, so it is fully interpretable and accessible. One can apply EN to a variety of applications with comprehensive understanding.

In our future work, we are going to tackle the computational cost of EN. Although we dealt with it by introducing the cascade with LDA in this paper. Introducing the fast template evaluation with vector quantization, that was recently proposed by Sadeghi et al.[10], is interesting for future work. Also, meta-data transfer using EN needs further development. We plan to modify the procedure of meta-data transfer for better accuracy.

References

[1] N. Dalal and B. Triggs, *Histograms of Oriented Gradients for Human Detection*, CVPR, 2005.

[2] M. Everingham, L. Gool, C. K. I. Williams, J. Winn, and A. Zisserman, *The Pascal Visual Object Classes (VOC) Challenge*, IJCV, vol.88, no.2, pp.303–338, 2009.

[3] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan, *Object detection with discriminatively trained part-based models*, PAMI, vol.32, no.9, pp.1627–1645, 2010.

[4] M. Gharbi, T. Malisiewicz, S. Paris, and F. Durand, *A gaussian approximation of feature space for fast image similarity*, MIT CSAIL Technical Report, 2012.

[5] C. Gu, P. Arbelaez, Y. Lin, K. Yu, and J. Malik, *Multi-component models for object detection*, ECCV, pp.445–458, 2012.

[6] B. Hariharan, J. Malik, and D. Ramanan, *Discriminative decorrelation for clustering and classification*, ECCV, pp.459–472, 2012.

[7] T. Malisiewicz, A. Gupta, and A. A. Efros, *Ensemble of exemplar-SVMs for object detection and beyond*, ICCV, pp.89–96, 2011.

[8] D. L. Medin and M. Schaffer, *Context theory of classification learning*, Psychological Review, pp.85:207–238, 1978.

[9] R. M. Nosofsky, *Attention, similarity, and the identification-categorization relationship*, Journal of Experimental Psychology: General, vol.115, no.1, pp.39–57, 1986.

[10] M. A. Sadeghi and D. Forsyth, *Fast Template Evaluation with Vector Quantization*, NIPS, 2013.

[11] X. Zhu, C. Vondrick, D. Ramanan, and C. Fowlkes, *Do We Need More Training Data or Better Models for Object Detection?*, BMVC, pp.80.1–80.11, 2012.