MULTIPLE INSTANCE LEARNING FOR MODEL ENSEMBLE AND META-DATA TRANSFER

Yu Chen[†], *Ling Cai*[⋆∗], *Yuming Zhao*[†], *Fuqiao Hu*[†]

[†]School of Electronic Information and Electrical Engineering, Shanghai Jiao Tong University
[⋆]School of Information Science and Engineering, Xiamen University

ABSTRACT

Traditional Exemplar-SVMs (ESVM) require millions of negative samples to establish a linear exemplar detector. However, Exemplar Linear Discriminant Analysis (ELDA) can achieve similar performance while avoid negative samples mining. To construct a strong object classifier, Multiple Instance Learning (MIL) is used to combine exemplar detectors and reduce annotation ambiguity. By applying MIL to Exemplar-LDA (ELDA), we simplify the training process and achieve better performance than ESVM on object detection. Moreover, exemplar models can transfer the available meta-data (segmentation, geometric structure, etc.) of training samples directly onto the detected objects, which provide more accurate and richer attributions than the detection results of a bounding box.

Index Terms— Linear Discriminant Analysis, Exemplar classifier, Multiple Instance Learning, Meta-data transfer

1. INTRODUCTION

Object detection has experienced a long time of development history and formed several classical frameworks, e.g., wavelet-based Adaboost detector, HOG and SVM classifier[1], Deep Learning[2, 3, 4], Deformable Part-based Models [5] and so on. Unfortunately, it is hard for these frameworks to interpret object attributes till ESVM[6, 7, 8] came into being. Typical object detection model just put a coarse bounding box around the object and assign a category tag, but ESVM aims to form a simultaneous association between the detected object and one single exemplar among the training samples. In this way, the exemplar meta-data (segmentation, geometric structure, 3D models, etc.) can be transferred with the detection results, which provide more detailed object attributions about the overall object understanding. That is to say, ESVM not only answers "What it is?" but also explains "What it is like?"[9, 10]. This distinguished characteristic of ESVM resembles much closely to our human being thinking pattern.

∗Corresponding author. This work is supported by NSFC 61175009 and NSFC 61572410.

The second critical characteristic of ESVM is that, instead of training a classifier for one class, it builds an individual linear detector for each exemplar of one training sample class. Therefore, there is no need to map various features to a common space during the exemplar-specific learning process. A linear SVM detector, learnt from only one positive exemplar and millions of negatives, is quite specific to its corresponding positive exemplar. Each ESVM objective function is convex and can be optimized independently. Then, a nearest-neighbor approach is used to aggregate these individual linear SVM detectors to form a monolithic linear SVM classifier for an object class, in order to achieve the detection performance comparable with much more complicated latent part-based model [11].

For constructing a linear SVM object classifier, ESVM has to search millions of negative samples for hard ones as its support vectors. Unfortunately, this procedure is quite time consuming and will be intractable as the number of categories increases. To overcome this obstacle, Hariharan et al. revisit a classical method, Linear Discriminant Analysis (LDA), to demonstrate ELDA models [12, 13] can be trained economically.

As usual, supervised learning methods require a large quantity of annotated training samples. But in some real environment, it is difficult to assign obvious labels to all training samples. Babenko proposes a learning paradigm called Multiple Instance Learning (MIL)[14, 15, 16] which allows ambiguously labeled data for training process. Tuchiya et al. present the Exemplar Network [17] to find the best possible mixture of specific exemplar-based detectors to form a generalized mixture model. It is also inspired by ELDA models and maintains meta-data transfer function. This general and concise network can be applied to various applications with comprehensive understanding.

2. OVERVIEW

Based on MIL and ELDA, our model makes use of ESVM fundamental framework as well as its intensive understanding of the scene. Firstly, we builds a discriminative linear LDA detector for each exemplar instead of a linear SVM detector. Then, we employ MIL to establish a monolithic classifier for

an object class on the basis of various separate LDA detectors. These two training processes make our model not only distinguish every specific exemplar but also identify an object class. Last but not least, it remains the intrinsic factor of exemplars meta-data (segmentation, geometric structure, etc.) transfer function.

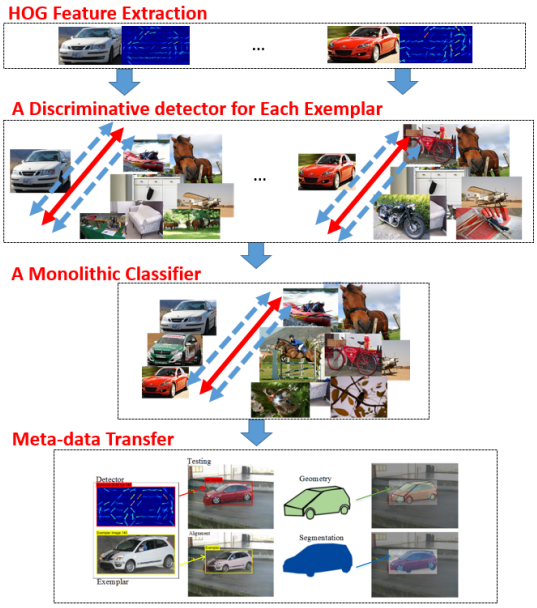

Fig. 1. Our model's main workflow of object detection.

3. MODEL LEARNING ALGORITHM

3.1. ELDA Classifier

A specific detector for each exemplar is trained from one single positive instance and millions of negatives. Typically, an exemplar-specific linear SVM detector has to mine a large amount of negative samples to find hard ones as support vectors. For one exemplar, its negative-mining process usually integrates numerous iterations and costs a long computaitonal time. At each iteration, many negative samples are mined to search for the support vectors. For L training samples (typically hundreds in PASCAL VOC 2007 for one category) and T mining iterations for each exemplar, the entire iteration requires $T * L$ times. Taking the time consumption of each iteration into account, the mining process for an object class is quite expensive.

On the contrary, a linear LDA detector does not need the mining iteration process, so that we can build a discriminant linear LDA detector for each exemplar to skip the costly mining process, and achieve similar detection performance.

Given the training samples $X = [x_1, x_2, \cdots, x_L]$ and the corresponding class label $y_i \epsilon \{-1, 1\}$ of x_i. x without the subscript represents the testing samples. A linear LDA detector can be described as follows:

$$f_{LDA}(x) = w_{LDA}^T \cdot x \tag{1}$$

where $w_{LDA} = \xi^{-1}(x_i - \mu_0)$, and x_i is the HOG feature vector extracted from each specific exemplar. It is closely related to the object class but has restored continuity of background contours. According to Eq.(1), LDA makes the positive feature x_i, centering it with μ_0 and whitening it with ξ^{-1}, which suppresses the contours of background.

The covariance ξ is computed over all training samples regardless of class labels. As Hariharan et al.[12] mentioned, the number of training samples belonging to an object class is usually small compared with the whole windows, so a generic and object independent μ_0 is enough. Therefore, the background template of μ_0 and ξ only need to be estimated once, and can be reused for all object classes.

The scale and translation invariance can further simplify dimension variance problem of the background template. From windows of different scales and translations, the smallest unit of negative vector μ_0 and covariance ξ can be obtained. We repeat them to the corresponding dimension of x_i for the LDA model. In this way, we generate a specific LDA detector for each exemplar from the training samples without hard negative mining.

3.2. MIL for monolithic classifier

As usual, the exemplar-based method will regulate scores of detection results from each exemplar with the nearest-neighbor approach[6], in order to get these scores comparable and preliminary classifiers associated.

However, the nearest-neighbor approach failed to figure out whether each detection window deserves its score or not. To further work it out , we construct a monolithic object classifier based on MIL paradigm to distinguish these detected windows without specific labels. As a supervised learning paradigm, MIL can construct a classifier from training samples with some ambiguous labels.To deal with ambiguously labeled samples, MIL trains the samples in the unit of bags and gives each bag a certain label. For the binary classification, the assumption in MIL is that a positive bag contains one positive sample at least, and a negative bag only contains negative samples. Thus, the training samples are the bags $\{X_1, X_2, \cdots, X_n\}$ and the bag labels $\{y_1, y_2, \cdots, y_n\}$, where $X_i = \{x_{i1}, x_{i2}, \cdots, x_{im}\}$, $X_i \epsilon \mathrm{X}$ and $y_i \epsilon \mathrm{Y}$. Generally, $\mathrm{X} = R^{\mathrm{d}}$ is the d-dimensional Euclidean space, and $\mathrm{Y} = \{0, 1\}$. $y_i' = 2y_i - 1 \in \{-1, 1\}$ can also be defined.The goal of MIL is to train a bag classifier $H(X_i) : \mathrm{X}^{\mathrm{m}} \rightarrow \mathrm{Y}$.

We formulate the objective function of MIL:

$$\min_{y_{ij}} \min_{\omega,b,\varepsilon} \frac{1}{2}\|w\|_2^2 + C\sum\nolimits_{ij}\varepsilon_{ij} \quad (2)$$
$$s.t. y_{ij} = 0, \forall i | y_i = 0$$
$$\sum\nolimits_j y_{ij} \geq 1, \forall i | y_i = 1$$
$$y'_{ij}(w \cdot x_{ij} + b) \geq 1 - \varepsilon_{ij}$$

where $\varepsilon_{ij} \geq 0$ is the slack variable for each point x_{ij}, because the data may not be separated absolutely. It attempts to recover the latent variable y_{ij} of every training sample, including negative samples in positive bags. Setting SVM classifier penalty coefficient $C_u = [C_u^+, C_u^-]$, use 10-fold cross validation approach to regulate it. Through the cross validation, each positive sample package and negative sample package have been taken as a validation sample and a training sample. Under a set of parameters, we will get 10 groups of average object detection rate $AP_i (i = 1, 2, \cdots, 10)$. Calculate the mean number of these ten APs, referred to as AAP, and compare values of AAPs under different penalty parameters. The penalty coefficient C_u and classifier $[w, b]$ corresponding to the maximum AAP value is the monolithic category classifier which MIL process finds. Once getting these variables, a linear SVM classifier $[w, b]$ for one category will be generated. A monolithic category classifier here can filter out testing results from individual exemplar detectors, increasing the average accuracy rate of object detection.

4. EXPERIMENTAL EVALUATION

We evaluate the object detection results of our model on the widely used benchmark dataset, PASCAL VOC 2007. The detection results are compared with that of ESVM model and ELDA model respectively. Moreover, we vividly present its ability of meta-data transfer which shows a good alignment between training exemplars(segmentations and geometries) and their related objects.

We select four categories of rigid bodies, cars, sofas, trains and aeroplanes from the benchmark samples to perform object detection and performance analysis. Under the same accuracy, the meta-data transfer effect of these categories is more obvious. Due to the fact that their meta-data are incomplete, we supplement them manually. Given these training samples, we construct an image pyramid for every single image and convert each layer of the image pyramid into the HOG space.

Experimental evaluation is conducted in two aspects: the object detection average accuracy rate (AP) and the corresponding Precision-Recall Curve. The average accuracy and the related parameter settings are listed in Table1 and Table2.

According to the statistical data in Table1, ELDA model achieves similar performances as ESVM model on the whole. It proves that a LDA detector can indeed replace a SVM detector during the specific exemplar training period, which saves lots of training time and maintains comparable performances. However, ELDA model cannot maintain stable outputs. For cars, it improves the accuracy but for trains it reduces the accuracy. By introducing MIL to ELDA models, these average precisions generated by our model, have increases of 26.4%, 25.2%, 7.6%, 21.9% respectively. The result shows that our model steadily increases the average detection accuracy of positive samples and simultaneously suppresses the false detection rate of negative samples.

Table 1. Experimental results

Category	Average Precision		
	ESVM	ELDA	Our model
car	66.8%	75.0%	**93.2%**
sofa	30.6%	33.4%	**55.8%**
train	58.1%	53.6%	**65.7%**
aeroplane	47.7%	49.1%	**69.6%**

Table 2. Sample parameter settings

Category	Detailed Setting Numbers			
	exemplars	Training samples	Test samples	Negative samples
car	500	261	259	500
sofa	248	229	223	500
train	297	261	259	500
aeroplane	306	238	204	500

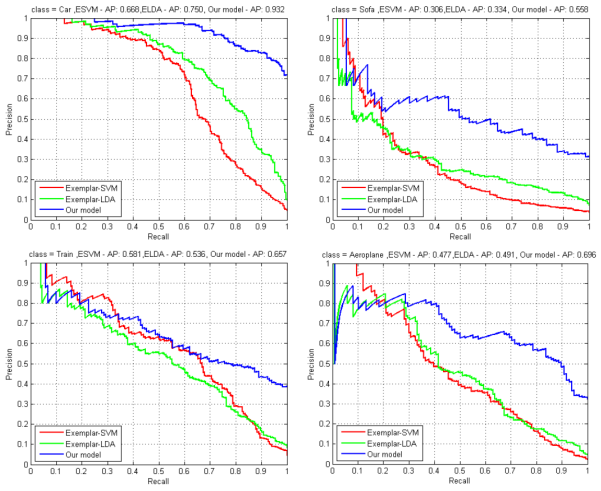

Fig. 2. The Precision-Recall curves of cars, sofas, trains and aeroplanes. ESVM, ELDA and our model correspond to the colors red, green, and blue.

In Fig.2, the red curves represent PR curves of ESVM in testing samples of cars, sofas, trains and aeroplanes. The

green ones and blue ones correspond ELDA model and our model respectively. For each category detection windows, setting a gradually reduced threshold to acquire different pricison and recall rates, which constitute the vertical and horizontal coordinates of points on the Precision-Recall curves. At the begining, a larger threshold leaves out few detection windows, most of which are accurate results. So the pricision is close to 1, but the recall is nearly 0. In the end, a smaller threshold brings numerous detection windows, most of which are wrong results. So the precision is nearly 0, but the recall is close to 1. Despite precision and recall have an inverse relationship, a high precision and a high recall at the same time are what we are attempting to pursuit. For each type of objects, the green curve fluctuates up and down near the red line. Overall, its effect is almost the same as the red one. On the other hand, the blue curve is basically above the red line and the blue one, which maintains a relatively slow downward trend. Thus, it can still maintains good accuracy at a high recall rate. In addition, because the linear LDA detector from ELDA model and our model leaves out negative samples mining, the training time that every exemplar consumes is greatly reduced by a few seconds, greatly improving the training speed.

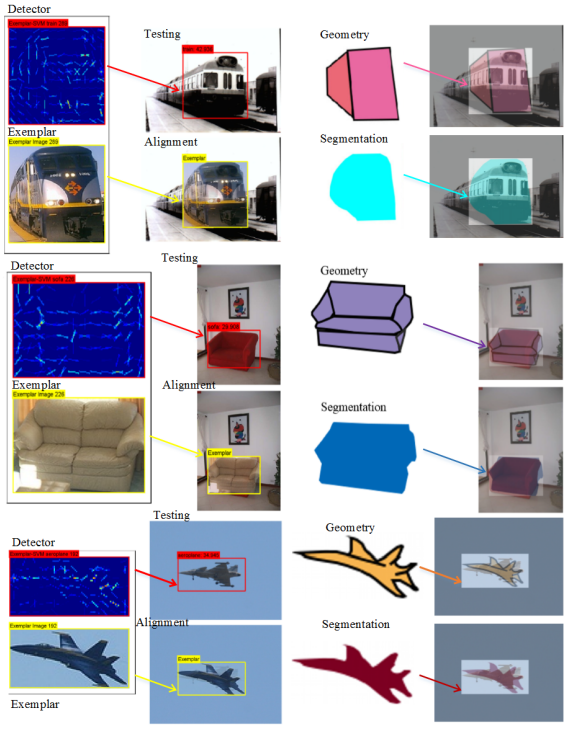

Fig. 3. The meta-data transfer performance of our model on the train and sofa and aeroplane.

With the high-quality alignment between detected objects

Fig. 4. The meta-data transfer performance of our model on various cars.

and exemplars, we can transfer the knowledge of exemplars (e.g., segmentation and geometry structure) directly onto object locations, to obtain shapes, directions and other high-level information. After sorting detected windows from high scores to low scores, we analysis the top ten detection results. As it is dipicted in Fig.3 and Fig.4, every detected window is found and replaced by its corresponding exemplar, which is discrimatively trained and keeps unique exemplar-based information. The exemplar and the transferred object maintain a high degree of consistency. Meanwhile, corresponding exemplars can also be its mete-data, such as geometry structures and segementations in Fig.3 and Fig.4. Therefore, combining the experimental result statistics with images, our model indeed not only tells us "what it is" but also reminds us of "what it is like".

5. CONCLUSION

Our ELDA model binding MIL not only leaves out negative samples mining during the training process, saving training time largely. Moreover, MIL is introduced to achieve the higher average accuracy in PASCAL VOC 2007 database than ESVM and ELDA model, which demonstrates the feasibility and effectiveness of our model. Meanwhile, the model retains the advantages of exemplar meta-data transfer function, and makes the object detection process more flexible, fully expressing the shape, orientation, size and other information of detected objects.

6. REFERENCES

[1] Navneet Dalal and Bill Triggs, "Histograms of oriented gradients for human detection," in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*. IEEE, 2005, vol. 1, pp. 886–893.

[2] Geoffrey E Hinton, Simon Osindero, and Yee-Whye Teh, "A fast learning algorithm for deep belief nets," *Neural computation*, vol. 18, no. 7, pp. 1527–1554, 2006.

[3] Geoffrey E Hinton and Ruslan R Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, no. 5786, pp. 504–507, 2006.

[4] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

[5] Pedro Felzenszwalb, David McAllester, and Deva Ramanan, "A discriminatively trained, multiscale, deformable part model," in *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*. IEEE, 2008, pp. 1–8.

[6] Tomasz Malisiewicz, Abhinav Gupta, Alexei Efros, et al., "Ensemble of exemplar-svms for object detection and beyond," in *Computer Vision (ICCV), 2011 IEEE International Conference on*. IEEE, 2011, pp. 89–96.

[7] Abhinav Shrivastava, Tomasz Malisiewicz, Abhinav Gupta, and Alexei A Efros, "Data-driven visual similarity for cross-domain image matching," in *ACM Transactions on Graphics (TOG)*. ACM, 2011, vol. 30, p. 154.

[8] Tomasz Malisiewicz, Abhinav Shrivastava, Abhinav Gupta, and Alexei A Efros, "Exemplar-svms for visual ob ject detection, label transfer and image retrieval.," in *ICML*, 2012.

[9] Tomasz Malisiewicz, *Exemplar-based representations for object detection, association and beyond*, Carnegie Mellon University, 2011.

[10] Mathias Eitz, James Hays, and Marc Alexa, "How do humans sketch objects?," *ACM Trans. Graph.*, vol. 31, no. 4, pp. 44, 2012.

[11] Pedro F Felzenszwalb, Ross B Girshick, David McAllester, and Deva Ramanan, "Object detection with discriminatively trained part-based models," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 32, no. 9, pp. 1627–1645, 2010.

[12] Bharath Hariharan, Jitendra Malik, and Deva Ramanan, "Discriminative decorrelation for clustering and classification," in *Computer Vision–ECCV 2012*, pp. 459–472. Springer, 2012.

[13] Aapo Hyvärinen, Jarmo Hurri, and Patrick O Hoyer, *Natural Image Statistics: A Probabilistic Approach to Early Computational Vision.*, vol. 39, Springer Science & Business Media, 2009.

[14] Boris Babenko, "Multiple instance learning: algorithms and applications," *View Article PubMed/NCBI Google Scholar*, 2008.

[15] Mohammad H Poursaeidi and O Erhun Kundakcioglu, "Robust support vector machines for multiple instance learning," *Annals of Operations Research*, vol. 216, no. 1, pp. 205–227, 2014.

[16] Yixin Chen, Jinbo Bi, and James Z Wang, "Miles: Multiple-instance learning via embedded instance selection," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 28, no. 12, pp. 1931–1947, 2006.

[17] Chikao Tsuchiya, Tomasz Malisiewicz, and Antonio Torralba, "Exemplar network: A generalized mixture model," in *Pattern Recognition (ICPR), 2014 22nd International Conference on*. IEEE, 2014, pp. 598–603.