The 7th Annual IEEE International Conference on
Cyber Technology in Automation, Control and Intelligent Systems
July 31-August 4, 2017, Hawaii, USA

A DPM Based Approach to Joint Object Detection and Sub-category Recognition

Liangji Fang, Tianwei Lin, Jun Wu, and Xu Zhao*

***Abstract*—Object detection and sub-category recognition play important roles in the field of computer vision. Most of the existing approaches separate detection and recognition into two sequential parts. We argue, however, detection and recognition could share information of each other to achieve a better performance for both of them. In this paper, a new approach to joint detection and recognition based on Deformable Part Model (DPM) is presented. Our approach extends DPM from pure object detection to simultaneous detection and sub-category recognition. A multi-objective optimization function is formulated. It integrates supervised sub-category recognition into DPM training process, using structural SVM with latent variables. The experiments show that our approach achieves a very exciting result in a challenging vehicle data set.**

I. Introduction

Computer vision is trying to answer the question of "what is where" on earth [1]. Object detection answers the question of "where", and recognition answers "what". Fast and accurate object detection and recognition are essential vision tasks and the basements of many other senior applications.

Object detection and sub-category recognition are likely to be separated into two parts in most of the existing methods, recognition following after detection. However, detection and recognition should not be separated completely. There are some information coupling between them. The process of detection can provide useful information for recognition. In return, recognition can help to confirm or reject the results of detection [2], [3]. Our idea is to combine detection and sub-category recognition into an integrated framework. In the joint approach, the coupling information can be fully used to accomplish object detection and sub-category recognition better simultaneously.

Our approach is mainly based on deformable part model (DPM) [4], [5]. DPM is a very useful approach to detect objects in static images. It is developed based on [6], [7]. And lots of researches are based on it in the recent years [8], [9]. DPM uses E-HOG (Enhanced HOG), which is based on HOG [10] but better than it. And latent SVM is used for model training.

Figure 1 shows an DPM model for bicycle. A complete DPM model is composed of several small models called components with different aspect ratios. Each component consists of three portions, a root filter, several part filters and a spatial model. Root filter describes objects in a rough manner, part filters in a detailed manner, and the spatial model is to describe the relative positions among root filter and part filters. The part filters and the components, offering lots of details about objects, provide the possibility of our extension.

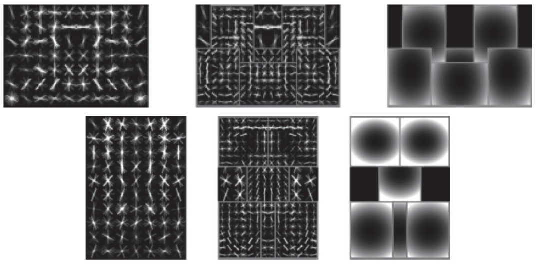


Fig. 1: A DPM model for bicycle with two components. Component in the first row captures sideways views of bicycle while the second row captures frontal views. Every component consists of three portions, a root filter, several part filters, and a spatial model [4] . Our idea is to make each component corresponding to one specific sub-category.

In this work, we integrate supervised sub-category recognition into DPM training process, making every component corresponding to one specific sub-category. It enables the model's ability of simultaneous object detection and sub-category recognition. In vehicle detection and recognition experiments, both recall and precision of detection can reach higher than 95%, and the accuracy of recognition is also higher than 95%. Compared with the original DPM, our approach does not spoil the detection ability and gains a strong recognition ability.

II. problem formulation

DPM uses sliding window scheme to detect objects in static images using feature pyramid. The model with several components is applied at every hypothetic position in the feature pyramid. Using x to stand for one example, we can get different features such as $\phi(x,z)$. $z \in Z$ means the latent variables described in [4], containing the choice of mixture component and parts deformation. Next, the model parameters $\boldsymbol{\beta}$ and the corresponding feature execute a convolution described as

$$score(x) = \max_{z \in Z} \boldsymbol{\beta} \cdot \boldsymbol{\phi}(x,z). \quad (1)$$

All authors are with the Key Laboratory of China MOE for System Control and Information Processing Department of Automation, Shanghai Jiao Tong University, Shanghai, China. This work has been supported by NSFC grants (61673269, 61375019).

* Contacting Author: Xu Zhao is with the Key Laboratory of China MOE for System Control and Information Processing Department of Automation, Shanghai Jiao Tong University, Shanghai, China. (email:zhaoxu@sjtu.edu.cn)

978-1-5386-0490-8/17/$31.00 © 2017 IEEE

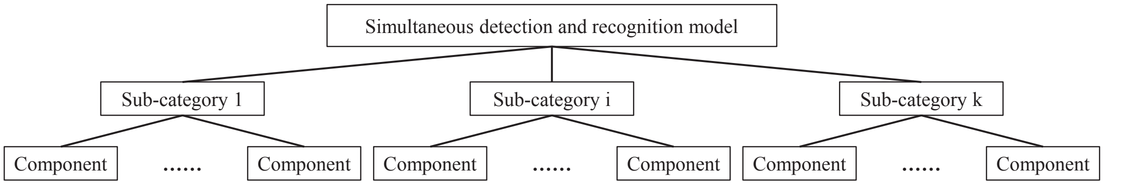


Fig. 2: Structure of our model for simultaneous object detection and sub-category recognition. The whole model is separated by sub-category difference into several portions. Every portion for one sub-category has several corresponding components. Components for the same sub-category are divided by aspect ratio as in original DPM.

Then, a threshold is chosen to pick up the scores which are bigger than it. Finally, non-maximum suppression is used to find out the most suitable detection rectangle.

Using $y^d \in \{+1, -1\}$ to stand for the label for object detection of example x and d denotes detection, the original objective function of DPM can be described as

$$L(\boldsymbol{\beta}) = \frac{1}{2}\|\boldsymbol{\beta}\|^2 + C\sum_{i=1}^{N} \max(0, 1 - y_i^d \max_{z \in Z} \boldsymbol{\beta} \cdot \boldsymbol{\phi}(x_i, z)). \quad (2)$$

It accomplishes supervised object detection in training.

The structure of our model is shown in Figure 2. We attempt to enable the trained model's ability of accomplishing simultaneous object detection and sub-category recognition based on DPM. While different components have different aspect ratios in DPM, our approach tries to make every sub-category has its own corresponding components.

Let y_i^r be the label for sub-category recognition, where r denotes recognition. We try to bring supervision of sub-category recognition into the training process. An simple idea is adding punishments such as 0-1 loss to the objective function of DPM directly to get

$$\begin{aligned} L(\boldsymbol{\beta}) = \frac{1}{2}\|\boldsymbol{\beta}\|^2 + C_1 \sum_{i=1}^{N} \max(0, 1 - y_i^d \max_{z \in Z} \boldsymbol{\beta} \cdot \boldsymbol{\phi}(x_i, z)) \\ + C_2 \sum_{y_i^d = +1} \Delta(y_i^r, y^r) \quad (3) \end{aligned}$$

where y^r is actually specified by z during the detection process and C_1, C_2 are the weight terms. In equation (1), once the latent variables z is specified, we can get the sub-category recognition result simultaneously. It's to say that $\hat{y}^r$ is corresponding to

$$\hat{z} = \operatorname*{argmax}_{z \in Z} (\boldsymbol{\beta} \cdot \boldsymbol{\phi}(x_i, z)). \quad (4)$$

However, it's very hard to do optimization on equation (3) directly. Because it's difficult to find out an explicit relationship between $\boldsymbol{\beta}$ and $\Delta(y_i^r, \hat{y}^r)$.

III. Multi-objective optimization

To get an explicit relationship, some modification should be made to equation (3). We can use

$$\begin{aligned} S_i(\boldsymbol{\beta}) = M\Big(\max_{y^r \in \{\textit{sub-category}\}, y^r \neq y_i^r} \big(\Delta(y_i^r, y^r) \\ + \max_{z \in Z} \boldsymbol{\beta} \cdot \boldsymbol{\phi}(x_i, z \mid y^r)\big) - \max_{z \in Z} \boldsymbol{\beta} \cdot \boldsymbol{\phi}(x_i, z \mid y_i^r)\Big) \quad (5) \end{aligned}$$

as a surrogate function to replace the loss of sub-category recognition, where $M(x) = \max(0, x)$ and $z \mid y^r$ stands for the latent variables with recognition label y^r. And it's easy to get $S_i(\boldsymbol{\beta}) \geqslant \Delta(y_i^r, \hat{y}^r)$.

Equation (5) comes from structural SVM but is a little different from the original one [11], [12]. There are some extra latent variables in equation (5) which makes it non-convex. It's actually a form of structural SVM with latent variables [13]. Subsequently, we can get the following multi-objective optimization function,

$$\begin{aligned} L(\boldsymbol{\beta}) = \frac{1}{2}\|\boldsymbol{\beta}\|^2 + C_1 \sum_{i=1}^{N} \max\big(0, 1 - y_i^d \max_{z \in Z} \boldsymbol{\beta} \cdot \boldsymbol{\phi}(x_i, z)\big) \\ + C_2 \sum_{y_i^d = +1} M\Big(\max_{y^r \in \{\textit{sub-category}\}, y^r \neq y_i^r} \big(\Delta(y_i^r, y^r) + \\ \max_{z \in Z} \boldsymbol{\beta} \cdot \boldsymbol{\phi}(x_i, z \mid y^r)\big) - \max_{z \in Z} \boldsymbol{\beta} \cdot \boldsymbol{\phi}(x_i, z \mid y_i^r)\Big). \quad (6) \end{aligned}$$

A. Semi-convexity

Equation (6) is non-convex. When $y_i^d = -1$, it's convex but not for $y_i^d = +1$. It's so-called semi-convexity. To solve this problem, we introduce CCCP [13], [14]. Once the features in the non-convex parts are specified, the two corresponding parts become linear. Then the overall function turns to be convex. In equation (6), for a given $\boldsymbol{\beta_t}$, we use

$$\hat{\boldsymbol{\phi}}^{\boldsymbol{d}}(x_i) = \boldsymbol{\phi}\big(x_i, \max_{z \in Z} \boldsymbol{\beta_t} \cdot \boldsymbol{\phi}(x_i, z)\big) \quad (7)$$

$$\hat{\boldsymbol{\phi}}^{\boldsymbol{r}}(x_i) = \boldsymbol{\phi}\big(x_i, \max_{z \in Z} \boldsymbol{\beta_t} \cdot \boldsymbol{\phi}(x_i, z \mid y_i^r)\big) \quad (8)$$

to replace the two non-convex parts, getting

$$\begin{aligned}\overline{L(\boldsymbol{\beta})} = \frac{1}{2}\|\boldsymbol{\beta}\|^2 &+ C_1 \sum_{y_i^d=-1} \max\left(0, 1 + \max_{z\in Z} \boldsymbol{\beta}\cdot\boldsymbol{\phi}(x_i, z)\right) \\ &+ C_1 \sum_{y_i^d=+1} \max\left(0, 1 - \boldsymbol{\beta}\cdot\hat{\boldsymbol{\phi}}^{\boldsymbol{d}}(x_i)\right) \\ + C_2 \sum_{y_i^d=+1} M\Big(&\max_{y^r\in\{\mathit{sub\text{-}category}\}, y^r\neq y_i^r} \left(\Delta(y_i^r, y^r) + \right. \\ &\left.\max_{z\in Z} \boldsymbol{\beta}\cdot\boldsymbol{\phi}(x_i, z \mid y^r)\right) - \boldsymbol{\beta}\cdot\hat{\boldsymbol{\phi}}^{\boldsymbol{r}}(x_i)\Big). \quad (9)\end{aligned}$$

Equation (9) then turns as convex and $\overline{L(\boldsymbol{\beta})} \geqslant L(\boldsymbol{\beta})$. It can be used as the surrogate function of equation (6). Then, standard convex optimization methods could be used, and we use L-BFGS, as used in DPM [15].

B. Cache policy

Considering about equation (9), there are two maximization operations among latent variables Z related with parameters $\boldsymbol{\beta}$. It's very time-consuming and not acceptable in practice. There is also a similar problem during the training process of DPM. Cache policy was introduced there [4]. DPM builds a cache C_d for object detection. Based on it, we introduce a similar cache named C_r for sub-category recognition.

Between two sequential optimization iterations, $\boldsymbol{\beta}$ changes in a small range and the features picked in the maximization operations in equation (9) should be near each other. So we can build caches to hold the features near the feature which gets the highest score at a specific $\boldsymbol{\beta_t}$. It reduces the space of latent variables to a great extent, in which the model searches for the highest response scores.

C_r could be built during the period of specifying surrogate function. Because they both require the model to be applied to the feature pyramids of positive examples for detection, searching for the corresponding features in each condition. C_d is built on negative examples as described in DPM.

C. Initialization

The multi-objective optimization function isn't convex. Though we've found out a way to do optimization on it, there's no guarantee that the globally optimal solution can be reached. It's somehow like EM algorithm. And it has a strong relationship with the initial parameters. So a careful initialization to the parameters is essential. We firstly use linear SVM to initialize every component with the corresponding sub-category examples, to get the initial parameters.

IV. Experiment

A. SJTUVehilce data set

To validate the performance of our approach, we collect about 270 thousands images about vehicles. About 5500 images are picked up to build a vehicle data set named SJTUVehicle. The location and sub-category information of the vehicles are annotated manually. We regard the vehicles as four different sub-categories, namely car, bus, minibus and truck. Details about the SJTUVehicle data set are shown in Table I.

TABLE I: Distribution of the SJTUVehicle data set.

Set	Pictures	Car	Minibus	Bus	Truck
Train	1827	905	518	236	430
Val	609	324	181	67	131
Test	3045	1575	837	391	668
Total	5481	2804	1536	694	1229

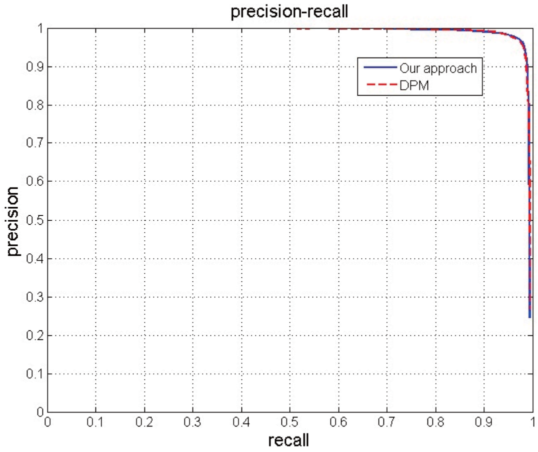


Fig. 3: Comparison of vehicle detection result between our approach and the original DPM.

B. Experimental setting

In our experiments, we pick the images from train set shown in Table I of SJTUVehicle data set as the positive examples and some images without vehicles in PASCAL VOC 2007 [16] as the negative. We try to detect vehicles in every image and classify each vehicle as car, bus, minibus or truck.

For every component, there are six part filters. For every sub-category, there are four components especially for it. There are sixteen components in total in our trained model.

C. Vehicle detection

We first test the vehicle detection ability of the newly trained model. The results are shown in Figure 3. The average precision is about 90.77%. In detail, the recall and precision can both reach higher than 95% with a suitable threshold. This result shows that our approach is effective on vehicle detection.

We trained an original DPM model with the same number of components and part filters as our newly trained model for vehicle detection. As shown in Figure 3, the two P-R curves are nearly the same. The average precision of vehicle detection of the original DPM model is about 90.77%, which is the same as our proposed approach. That's to say, our extension does not spoil detection ability of the original DPM model.

D. Vehicle recognition

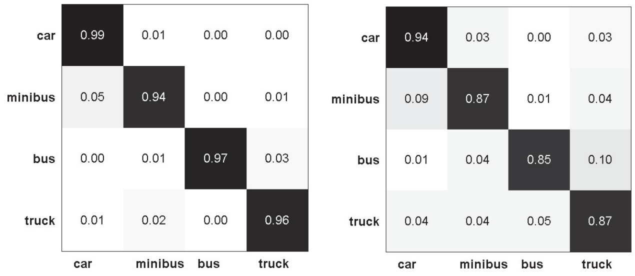
We also test the vehicle recognition ability. Because different recognition results can be obtained with different thresholds, so firstly we use F-Measure[1] of vehicle detection to find a proper threshold on validation set. When threshold is set as 0.15, F-Measure reaches the biggest. That means the precision and recall of vehicle detection achieve balance to some degree. Then we evaluate vehicle recognition results based on this threshold. Only the detected vehicles which are true positives, are used to evaluate the recognition ability, as there are no sub-category information for the false positives.

When the threshold is set as 0.15, vehicle recognition results are shown in Table II and the confusion matrix is shown in Figure 4(a). The overall accuracy of vehicle recognition is about 97.08%. The precision of vehicle detection is about 97.36% and the recall is about 96.83%. We show the examples of vehicle detection and recognition in Figure 5.

Using a linear SVM combined with HOG to do vehicle recognition, the overall accuracy is about 89.77%. And the confusion matrix is shown in Figure 4(b). It's obvious that our approach is much better.

TABLE II: Vehicle recognition result.

Type	Accuracy	Precision	Recall	F-Measure
Car	0.9824	0.9693	0.9928	0.9809
Minibus	0.9774	0.9673	0.9390	0.9529
Bus	0.9946	0.9837	0.9678	0.9757
Truck	0.9872	0.9717	0.9612	0.9664



(a) Our approach

(b) HOG+SVM

Fig. 4: Confusion matrix of vehicle recognition

V. Discussion

Our proposed approach extends DPM's detection to simultaneous detection and sub-category recognition. A new multi-objective optimization problem is introduced and structural SVM with latent variables is used. Detailed method to solve the problem is also given by strict mathematical analysis. The experiment on vehicle shows our approach doesn't spoil the detection ability of original DPM and enable the model a strong sub-category recognition ability.

The feature used in our experiment is E-HOG, and it seems not suitable for every task. So, it could be replaced by features from deep learning to gain a better performance in the future.

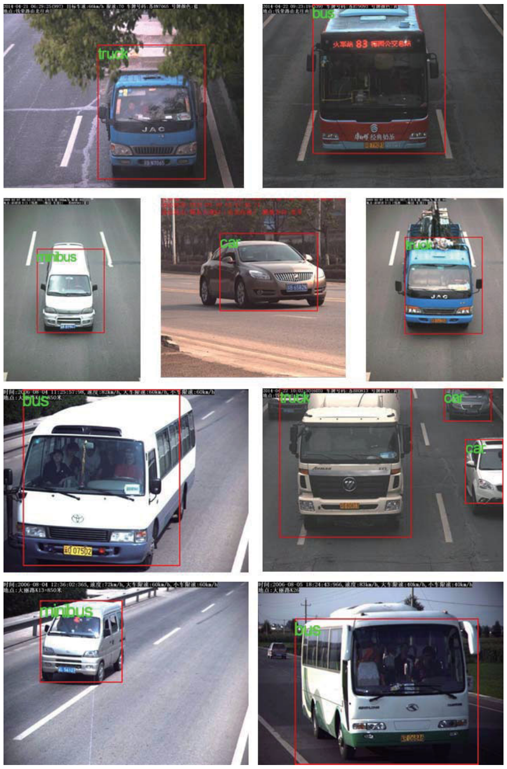


Fig. 5: Examples of vehicle detection and recognition result. The red rectangle stands for the detection result and the green text stands for the label from vehicle recognition.

References

[1] David Marr, *Vision: A Computational Investigation into the Human Representation and Processing of Visual Information*, Henry Holt and Co., Inc., Baltimore, Maryland, 1982.

[2] Hedi Harzallah, Frédéric Jurie, and Cordelia Schmid, "Combining efficient object localization and image classification," in *Computer Vision, 2009 IEEE 12th International Conference on*, Kyoto, Japan, 2009, IEEE, pp. 237–244, IEEE.

[3] Zheng Song, Qiang Chen, Zhongyang Huang, Yang Hua, and Shuicheng Yan, "Contextualizing object detection and classification," in *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, Colorado Springs, CO, 2011, IEEE, pp. 1585–1592, IEEE.

[4] 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.

[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*, Anchorage, AL, 2008, IEEE, pp. 1–8, IEEE.

[6] Martin A Fischler and Robert A Elschlager, "The representation and matching of pictorial structures," *IEEE Transactions on computers*, , no. 1, pp. 67–92, 1973.

[7] Pedro F Felzenszwalb and Daniel P Huttenlocher, "Pictorial structures for object recognition," *International Journal of Computer Vision*, vol. 61, no. 1, pp. 55–79, 2005.

[8] Pierre-André Savalle, Stavros Tsogkas, George Papandreou, and Iasonas Kokkinos, "Deformable part models with cnn features," in *European Conference on Computer Vision, Parts and Attributes Workshop*, Zurich, Switzerland, 2014, Springer.

[9] Ross B Girshick, Pedro F Felzenszwalb, and David A Mcallester, "Object detection with grammar models," in *Advances in Neural Information Processing Systems*, Granada, Spain, 2011, pp. 442–450, Curran Associates, Inc.

[1] F-Measure $= \frac{2PR}{P+R}$,where P stands for the precision of detection and R stands for the recall of detection.

[10] 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.

[11] Ioannis Tsochantaridis, Thomas Hofmann, Thorsten Joachims, and Yasemin Altun, "Support vector machine learning for interdependent and structured output spaces," in *Proceedings of the twenty-first international conference on Machine learning*, NY, USA, 2004, ACM, p. 104, ACM.

[12] Andrea Vedaldi, "Flexible discriminative learning with structured output support vector machines," Tech. Rep., University of Oxford, Oxford,England, 2013.

[13] Chun-Nam John Yu and Thorsten Joachims, "Learning structural svms with latent variables," in *Proceedings of the 26th Annual International Conference on Machine Learning*, Montreal, Quebec, Canada, 2009, ACM, pp. 1169–1176, ACM.

[14] Alan L Yuille, Anand Rangarajan, and AL Yuille, "The concave-convex procedure (cccp)," *Advances in neural information processing systems*, vol. 2, pp. 1033–1040, 2002.

[15] Ross Brook Girshick, *From rigid templates to grammars: Object detection with structured models*, University of Chicago, Chicago,USA, 2012.

[16] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes (voc) challenge," *International Journal of Computer Vision*, vol. 88, no. 2, pp. 303–338, June 2010.