Multiclass Object Detection by Combining Local Appearances and Context*

LiMin Wang[1,2], Yirui Wu[1], Tong Lu[1,*] and Kang Chen[1]
[1]State Key Lab for Novel Software Technology, Nanjing University, China
[2]Department of Information Engineering, The Chinese University of Hong Kong, Hong Kong
{07wanglimin,chenkangnobel}@gmail.com,wuyirui1989@163.com,lutong@nju.edu.cn

ABSTRACT

In this paper, we present a novel approach for multiclass object detection by combining local appearances and contextual constraints. We first construct a multiclass Hough forest of local patches, which can well deal with multiclass object deformations and local appearance variations, due to randomization and discrimination of the forest. Then, in the object hypothesis space, a new multiclass context model is proposed to capture relative location constraints, disambiguating appearance inputs in multiclass object detection. Finally, multiclass objects are detected with a greedy search algorithm efficiently. Experimental evaluations on two image data sets show that the combination of local appearances and context achieves state-of-the-art performance in multiclass object detection.

Categories and Subject Descriptors

I.4.8 [**IMAGE PROCESSING AND COMPUTER VISION**]: Scene Analysis

General Terms

Algorithms Experimentation

Keywords

Object Detection, Multiclass Hough Forest, Context Model

1. INTRODUCTION

Category-level object detection, i.e. predict the bounding boxes of object instances of a class in a test image, has been one of the most active areas in multimedia and computer vision. It has numerous applications such as driver assistance for automobiles by detecting pedestrians [5] and digital camera auto focus by using face detection [14]. There are two leading approaches to solve this problem: sliding windows [5, 14] and Hough voting [7, 10, 11]. Sliding windows scan over possible locations and scales, evaluate a classifier and use post processing to detect objects. Unfortunately, this procedure is a time-consuming work. Instead, Hough voting parametrizes object hypothesis and lets each local part vote for object centroid in the hypothesis space, greatly improving the detection efficiency. As a result, Hough voting has been successfully adapted to the problem of part-based object detection and obtained state-of-the-art results on some popular data sets in the past years. In [10], Leibe *et al.* introduce an *implicit shape model*(ISM), whose object part model is a set of visual words obtained generatively by clustering primitive image features. At running time, the interest point descriptors are matched against the visual words and each matching entry casts probabilistic votes about possible positions of the object in the scale space. In [7], Gall *et al.* present a discriminative Hough voting method called *Hough forest* for object detection. Rather than using a generative visual words, they learn a direct mapping between the appearance of an image patch and its Hough votes for object location.

Unlike single class object detection, there are less methods proposed for multiclass object detection [11, 6]. In [11], Opelt *et al.* present an incremental learning framework by combining shape features and local appearances for the problem of multiclass object detection. However, they ignore the contextual information, i.e. global scene statistics [13] or local interactions among objects [8] in the real world scene, which plays an important role in multiclass object detection. In [6], Desai *et al.* introduce an unified model for multiclass object detection that casts the problem as a structured prediction task. Their model learns statistics which capture the spatial arrangements of various object classes in real images. However, their method is still based on sliding windows and learns a window template for each object class, which cannot deal with object deformations and local appearance variations well .

In this paper, we present a novel approach for multiclass object detection by combining local appearances and context. We construct a multiclass Hough forest, which can efficiently model object deformations and local appearance variations. Meanwhile, contextual information, mainly the relative locations of different object classes, is incorporated into the multiclass Hough forest framework to disambiguate appearance inputs. Finally, we use a greedy search algorithm in the hypothesis space for multiclass object detection. Since our model considers both local appearances and object

[*] Area chair: Bernard Merialdo

MM'11, November 28–December 1, 2011, Scottsdale, Arizona, USA.
Copyright 2011 ACM 978-1-4503-0616-4/11/11 ...$10.00.

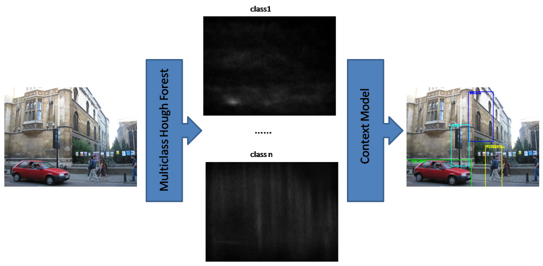


Figure 1: The detection processes of our approach.

interaction contexts, the greedy search algorithm efficiently improves the detection accuracy.

2. OUR APPROACH

We present a novel approach for multiclass object detection by combining local appearances and context. First, we establish a multiclass Hough forest and use local patches to vote for the possible locations of different objects. Then, we consider relative location constraints among objects in the Hough voting space and improve detection accuracy (See Fig. 1 for illustration).

2.1 Multiclass Hough Forest

Based on the single class Hough Forest proposed in [7], we introduce a multiclass Hough forest which can be used for multiclass object detection. Multiclass Hough forest is different from the single class Hough forest both in the training and detection processes. First, during the training phase, we train our forest by simultaneously modeling multiple object categories and thus different object classes can share common features. Second, in the detection phase, multiclass objects of interest can be directly detected by a single round of scanning over the image with our multiclass Hough forest.

Training data and leaf information. For our multiclass Hough forest, each tree $\mathcal{T}$ is constructed based on a set of patches from multiple object categories: $\{\mathcal{P}_i = (\mathcal{I}_i, c_i, \mathbf{d}_i)\}$, where $\mathcal{I}_i$ is the appearance of a local patch, c_i is the class label and $c_i \in \{0, 1, \ldots, |C|\}$, $\mathbf{d}_i$ is the offset of the patch. The training patches are randomly sampled from the training image collection and some of them contain examples of the class of interest with known bounding boxes. The patches from the background are assigned the class label $c_i = 0$ and the ones from interest objects are assigned class labels ranging from 1 to $|C|$, where $|C|$ is the total number of object classes. Every object patch is assigned a 2D offset vector $\mathbf{d}_i$, indicating the relative location of the patch from the object centroid.

For each leaf node L in the trees, the information about patches reaching this node is stored. We first introduce a portion list P_L to remember the portions of different class patches: $P_L = \{p_0, p_1, \ldots, p_{|C|}\}$, where p_i is the portion of the patches labeled with class i. Then we construct an offset matrix D_L to store the offsets of different class patches: $D_L = \{\mathbf{d}_{ij}\}$, where $\mathbf{d}_{ij}$ is the j^{th} patch offset labeled with class i. During the detection, the information is used to cast probabilistic Hough votes about the existence of the object at different positions (see **Detection over scales**).

Multiclass tree construction. The construction of our multiclass Hough forest follows the common random forest framework [4]. Each of the tree is constructed recursively from the root until the stopping conditions are satisfied, e.g. the depth of a node is equal to maxima(d_{max}) or the number of the patches on a fixed node is relatively small(N_{min}). There are two key parts during the construction: binary test and evaluation of the binary test quality. We use the same features and binary test with the single Hough forest construction (see details in [7]). Meanwhile, we adapt the following two uncertainties for a set of patches $A = \{\mathcal{P}_i = (\mathcal{I}_i, c_i, \mathbf{d}_i)\}$ to our multiclass Hough forest to evaluate the binary test quality: class-label uncertainty and offset uncertainty. The class label uncertainty is defined as follows:

$$U_1(A) = |A| \times \sum_{i=0}^{|C|} p_i log \frac{1}{p_i} \tag{1}$$

where $|A|$ is the size of set A and p_i is the probability of class i. Then the offset uncertainty is defined as :

$$U_2(A) = \sum_{i=1}^{|C|} \sum_{j:c_j=i} (\mathbf{d}_j - \mathbf{d}_{Ai})^2 \tag{2}$$

where $\mathbf{d}_{Ai}$ is the mean offset vector over all the patches labeled with i. Finally, the split criteria is the same to the single Hough forest and we pick the binary test t^k with the minimal sum of the respective uncertainty:

$$\arg\min_k \Big(U_*(\{\mathcal{P}_i | t^k(\mathcal{I}_i) = 0\}) + U_*(\{\mathcal{P}_i | t^k(\mathcal{I}_i) = 1\}) \Big) \tag{3}$$

where $* = 1$ or 2 (depending on the random choice).

Detection over scales. During the detection phase, we use the leaf information of our multiclass Hough forest to vote for different class object centroids in the hypothesis space. Consider a patch $\mathcal{P}(\mathbf{y}) = (\mathcal{I}(\mathbf{y}), c(\mathbf{y}), \mathbf{d}(\mathbf{y}))$ centered at position $\mathbf{y}$. We are now interested in the probabilistic vote $p(E(\mathbf{x}), E(i)|\, \mathcal{I}(\mathbf{y}))$, which means the appearance $\mathcal{I}(\mathbf{y})$ of a patch casts for the possibility of detecting an object of class i in position $\mathbf{x}$. For a single tree $\mathcal{T}$, we define the probability as follows:

$$p(E(\mathbf{x}), E(i)|\, \mathcal{I}(\mathbf{y}); \mathcal{T}) = \sum_{j:\mathbf{d}_{ij} \in D_L} \frac{1}{2\pi\sigma^2} \exp\left\{ -\frac{\| (\mathbf{y} - \mathbf{x}) - \mathbf{d}_{ij} \|^2}{2\sigma^2} \right\} \times \frac{p_i}{|D_L^i|} \tag{4}$$

where $|D_L^i|$ is the size of i^{th} row of matrix D_L. For the whole forest $\{\mathcal{T}_t\}_{t=1}^{T}$, we just average the probabilistic vote coming from different trees.

To deal with object scale variations, we resize a test image to a scale space and detect by multiclass Hough forest in each scale. Finally, we use the mean shift to find the maxima in the scale space.

2.2 Context Model

Inspired by the work of Galleguillos *et al.* [8], we model relative location constraints among multi-class objects in the hypothesis space generated by the Hough voting and incorporates the spatial context to our multiclass Hough forest.

Modeling object relations. We use a subset of the LabelMe data set [12] for the training and test in our context model. The subset is mainly composed of street scenes. Like [6], we classify object pairwise relationships into five groups: *ontop*, *far*, *next to*, *above* and *below* (See Fig.2). For each object pairwise relationship $R^i (i = 1, 2, 3, 4, 5)$, we define a

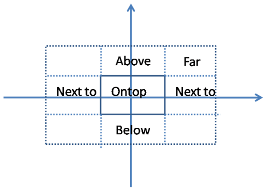


Figure 2: The location relationship between objects.

relationship matrix Φ^i, which captures the probability distribution of object class pairwises. The probability of an object class pairwise $< m, n >$ $(m, n = 1, \cdots, |C|)$ in the location relationship R^i is defined as follows:

$$p(< m, n >; \Phi^i) = \frac{1}{Z(\Phi^i)} \exp\{\phi^i_{mn}\} \tag{5}$$

where Φ^i is the relationship matrix and ϕ^i_{mn} is the entry (m, n) in the relationship matrix. $Z(\Phi^i)$ is the partition function. An object class pairwise $< m, n > \in R^i$ means two objects of class m and n satisfy the i^{th} relative location constraint illustrated in Fig.2. Thus given a data set D, the probability of relationship R^i is :

$$p(D; \Phi^i) = \frac{1}{Z(\Phi^i)^{M^i}} \exp\{\sum_{m=1}^{|C|} \sum_{n=1}^{|C|} l^i_{mn}\phi^i_{mn}\} \tag{6}$$

where l^i_{mn} is the entry (m, n) of a frequency matrix for spatial relationship R^i in the data set D, which counts the times object pairwise $< m, n >$ appears in a training image, satisfying relationship R^i, and M^i is the total number of object pairwise of relationship R^i.

Learning model parameters. Given a data set D, we wish to find Φ^i maximizes the log likelihood of the observed object pairwise:

$$\mathcal{L}(\Phi^i) = \log p(D; \Phi^i) = \sum_{m=1}^{|C|} \sum_{n=1}^{|C|} l^i_{mn}\phi^i_{mn} - M^i \times \log Z(\Phi^i) \tag{7}$$

Since we must evaluate the partition function, maximizing the log likelihood is intractable. Like [8], we approximate the partition function using Monte Carlo integration. The importance sampling is used and the proposal distribution is equal to their observed frequency. Thus we can use the gradient descent to find Φ^i, which approximately optimizes the likelihood, and the gradient is as follows:

$$\nabla_{\Phi^i}\mathcal{L}(\Phi^i) = \begin{bmatrix} l^i_{11} & \cdots & l^i_{1|C|} \\ \vdots & \ddots & \vdots \\ l^i_{|C|1} & \cdots & l^i_{|C||C|} \end{bmatrix} \tag{8}$$

Due to the noise of estimating the partition function, it is difficult to check for the convergence and we adopt the same trick with [8]: training is terminated when 10 iterations of gradient decent go not yield averagely improved likelihood over the previous 10.

Greedy search method. Based on the candidates, i.e. some local maxima, in the object hypothesis space produced by the multiclass Hough forest, we propose a greedy search algorithm, which well combines local appearances and context for multiclass object detection. First, we define the probability $P(\mathbf{x}, c)$ as:

$$P(\mathbf{x}, c) = \alpha P_{app}(\mathbf{x}, c) + (1 - \alpha)P_{con}(\mathbf{x}, c) \tag{9}$$

where $P(\mathbf{x}, c)$ is the probability that an object of class c is present at position $\mathbf{x}$, $P_{app}(\mathbf{x}, c)$ indicates the evidence from local appearances and $P_{con}(\mathbf{x}, c)$ means the one from contextual information, α is a weight factor between appearances and context. For each candidate $(\mathbf{x}_i, c_i)$, the probability $P_{app}(\mathbf{x}_i, c_i)$ is calculated according to the Hough vote results and the probability $P_{con}(\mathbf{x}_i, c_i)$ is the average of the object pairwise probability $p(< c_i, j >)$ with other objects. Then, we propose a greedy search algorithm to find object class labels for the candidates (see Table 1. for details).

Table 1: Greedy Search Algorithm.

Algorithm Greedy search for multiclass object detection by combining appearances and context.

Input. Some candidates: $(\mathbf{x}_1, c_1), \cdots, (\mathbf{x}_n, c_n)$ with the appearance probability: $P_{app}(\mathbf{x}_1, c_1), \cdots, P_{app}(\mathbf{x}_n, c_n)$.
Setp1. Set $R = \emptyset$ and initialize the context probability $P_{con}(\mathbf{x}_1, c_1), \cdots, P_{con}(\mathbf{x}_n, c_n)$ to be zero.
Step2. Search for $(\mathbf{x}_*, c_*) = \arg\max_{(\mathbf{x}, c_i) \notin R} P(\mathbf{x}_i, c_i)$. If $P(\mathbf{x}_*, c_*) > \theta$, $R = R \cup \{(\mathbf{x}_*, c_*)\}$, where θ is the detection threshold. Else stop.
Step3. For the remaining candidates, update the context probability:

$$P_{con}(\mathbf{x}_j, c_j) = \frac{1}{M}\left[(M-1)P_{con}(\mathbf{x}_j, c_j)) + \frac{\exp\{\phi^\star_{c_j c_*}\}}{Z(\Phi^\star)}\right]$$

where M is number of detecting objects and $\star \in \{1, \cdots, 5\}$. Go to **Step2**.
Output. The detection result $R = \{(\mathbf{x}_i, c_i)\}$.

3. EXPERIMENTS

9 classes data set. We first collect a data set of 9 classes: Face, Plane, Motorbike, CarRear [9], CarSide [1], TUD pedestrian [2], CowSide [10], Weizmann Horse [3] and Bottles (from Google Image) to evaluate the detection accuracy of multiclass Hough forest. Each class has 200 training images and 100 test images. We extract 50 patches from each training image and train a multiclass Hough forest composed of 15 trees with $d_{max} = 20$ and $N_{min} = 20$. Then we use the multiclass Hough forest to detect the 9 classes objects in test images. See Fig. 3 for the detection results($precision = \frac{num\ of\ right\ detections}{num\ of\ total\ detections}, recall = \frac{num\ of\ right\ detections}{num\ of\ total\ instances}$). It is very encouraging that the detection precision is averagely higher than 0.8 for the 9 classes objects and even achieves a higher rate for some rigid object classes such as CarSide and Motobike. It indicates that the multiclass Hough forest is robust and efficient for multiclass object detection.

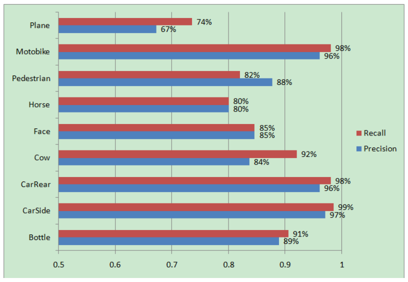
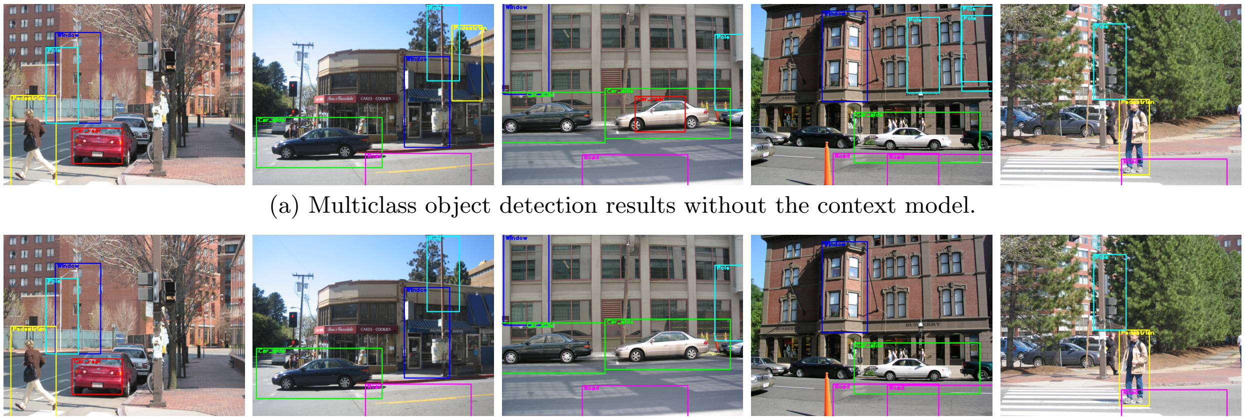


Figure 3: The detection result in 9 classes data set.



(a) Multiclass object detection results without the context model.

(b) Multiclass object detection results with the context model.

Figure 4: Examples of multiclass detection result in LabelMe data set.

Subset of LabelMe. Then, we use a subset of LabelMe data set [12] which is composed of 500 training images and 100 test images to evaluate whether our method is efficient for real world scenes. The subset is mainly composed of street scenes and we consider 6 class objects of interest in the scenes: Window, CarRear, CarSide, Pedestrian, Pole and Road. We use the object pairwise frequency matrix of the subset to train our context model. During the detection phase, we conduct two kinds of experiments: detection without context model and detection with context model. See Fig. 4 and Fig. 5 for the detection results. From the experiments, we find our multiclass Hough forest can accurately detect some rigid classes of objects from real scenes such as: Window, CarSide and Road. Meanwhile, from the left confusion matrix in Fig. 5, we find there exist wrongly detected instances or missed classes, such as Pedestrian and Pole. This is caused by the great variations or high deformations of the objects of these classes. After incorporating object relative location constraints with the appearances, we can successfully avoid such wrong detections and effectively decrease the false positive rate. The right confusion matrix in Fig. 5 illustrates our analysis.

One shortcoming of our method is the context model can now only help avoid wrong results after multiclass Hough forest detection, but can not improve the voting accuracy of our multiclass Hough forest (e.g. the missed instances due to occlusion in the last column in Fig. 4). The main reason is that our proposed greedy search algorithm is based on the local maxima in the hypothesis space and thus we can not find new instances if the votes are small. In the future work, we may consider combine our Hough forest and the context model into an unified model to solve the above problem.

4. ACKNOWLEDGEMENTS

The work was supported by Natural Science Foundation of China under Grant No. 60723003, 61021062 and 61003113, 973 Program of China under Grant No. 2010CB327903, and Natural Science Foundation of Jiangsu under Grant No BK 2009082.

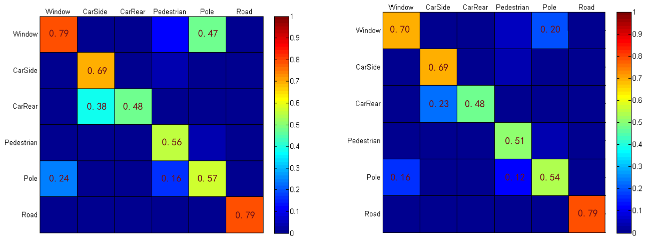


Figure 5: The confusion matrix for 6 classes. Left: results without context; Right: results with context.

5. REFERENCES

[1] S. Agarwal, A. Awan, and D. Roth. Learning to detect objects in images via a sparse, part-based representation. *TPAMI*, 26(11):1475 –1490, 2004.

[2] M. Andriluka, S. Roth, and B. Schiele. People tracking by detection and people detection by tracking. In *CVPR*, pages 1 – 8, 2008.

[3] E. Borenstein and S. Ullman. Learning to segment. In *ECCV*, pages 315–328, 2004.

[4] L. Breiman. Random forests. *Machine Learning*, 45(1):5–32, 2001.

[5] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *CVPR*, pages 886–893, 2005.

[6] C. Desai, D. Ramanan, and C. Fowlkes. Discriminative models for multi-class object layout. In *ICCV*, pages 229–236, 2009.

[7] J. Gall and V. S. Lempitsky. Class-specific hough forests for object detection. In *CVPR*, pages 1022–1029, 2009.

[8] C. Galleguillos, A. Rabinovich, and S. Belongie. Object categorization using co-occurrence, location and appearance. In *CVPR*, pages 1–8, 2008.

[9] G. Griffin, A. Holub, and P. Perona. Caltech-256 object category dataset. Technical Report 7694, 2007.

[10] B. Leibe, A. Leonardis, and B. Schiele. Robust object detection with interleaved categorization and segmentation. *IJCV*, 77(1-3):259–289, 2008.

[11] A. Opelt, A. Pinz, and A. Zisserman. Learning an alphabet of shape and appearance for multi-class object detection. *IJCV*, 80(1):16–44, 2008.

[12] B. C. Russell, A. Torralba, K. P. Murphy, and W. T. Freeman. Labelme: A database and web-based tool for image annotation. *IJCV*, 77(1-3):157–173, 2008.

[13] A. Torralba. Contextual priming for object detection. *IJCV*, 53(2):169–191, 2003.

[14] P. A. Viola and M. J. Jones. Robust real-time face detection. *IJCV*, 57(2):137–154, 2004.