Active Mask Hierarchies for Object Detection

Yuanhao Chen[1], Long (Leo) Zhu[2], and Alan Yuille[1]

[1] Department of Statistics, UCLA
[2] CSAIL, MIT

Abstract. This paper presents a new object representation, Active Mask Hierarchies (AMH), for object detection. In this representation, an object is described using a mixture of hierarchical trees where the nodes represent the object and its parts in pyramid form. To account for shape variations at a range of scales, a dictionary of masks with varied shape patterns are attached to the nodes at different layers. The shape masks are "active" in that they enable parts to move with different displacements. The masks in this active hierarchy are associated with histograms of words (HOWs) and oriented gradients (HOGs) to enable rich appearance representation of both structured (eg, cat face) and textured (eg, cat body) image regions. Learning the hierarchical model is a latent SVM problem which can be solved by the incremental concave-convex procedure (iCCCP). The resulting system is comparable with the state-of-the-art methods when evaluated on the challenging public PASCAL 2007 and 2009 datasets.

1 Introduction

The difficulty of object detection is because objects have complex appearance patterns and spatial deformations which can all occur at a range of different scales. Appearance patterns can be roughly classified into two classes: (i) structural (e.g., the head of a cat) which can be roughly described by the intensity edges and their spatial relations (e.g. by histogram of oriented gradients (HOGs)), and (ii) textural (e.g., the fur of a cat) which can be modeled by histograms of image features or words (e.g., histogram of words (HOWs)). Moreover these patterns can deform spatially both by translation – i.e., an entire image patch move – and/or by being partially masked out. The approach in this paper develops a novel Active Mask Hierarchy (AMH) which combines both types of appearance cues (HOWs and HOGs), allows subparts of the object to move actively and use a variety of different masks to deal with spatial deformations, and represents these appearance and geometric variations at a range of scales by a hierarchy.

Our work relates to two recent object representations which have made a significant impact in computer vision: (i) spatial pyramids [1], and (ii) part-based model [2]. But both approaches have strengths and weaknesses in the way that they deal with appearance variations and shape deformations. In this paper we seek an object representation which combines their strengths.

K. Daniilidis, P. Maragos, N. Paragios (Eds.): ECCV 2010, Part V, LNCS 6315, pp. 43–56, 2010.

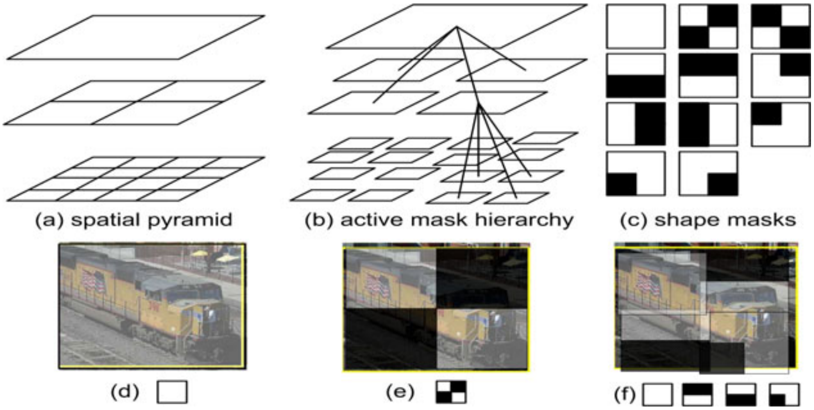


Fig. 1. (a) A spatial pyramid where the cells are bound together. (b) An Active Mask Hierarchy is represented by a tree structure where nodes are connected between consecutive layers and allowed to move object parts with displacements at different scales. (c) Shape masks include 11 generic shapes, such as vertical and horizontal bars, oriented L's, etc. The white regions show the "valid" areas, where features are computed, while the black regions are "invalid". The first mask is the rectangle used in standard spatial pyramids. (d) The train image example. A rectangle is used at the top layer to represent the entire object including some background. (e) Another (diagonal) mask is also used at the top layer to describe the train. (f) Four masks at the second level can be translated actively to better describe the object shape.

Spatial pyramids were proposed in [3] for scene classification and applied to object detection by [1]. A spatial pyramid is a three-layer pyramid, as shown in figure 1.(a), where cells at different levels of the grid specify histograms of words (HOWs) located in the corresponding spatial domain yielding a coarse-to-fine representation. HOWs are particularly successful at modeling textured regions (e.g., a cat's body), but are not well suited for describing structured regions (e.g., a cat's face). Some papers [4,1] use complementary descriptors, ie, histograms of oriented gradients (HOGs) [5], to account for other appearance variations. But two limitations still remain in the pyramid framework: (i) the cells are tightly bound spatially and are not allowed to move in order to deal with large spatial deformations of object parts (although pyramid of HOWs do tolerate a certain amount of spatial deformation). (ii) the cells have a rigid rectangular form and so are not well suited for dealing with partial overlaps of the object and its background. For example, the bounding box for the train in figure (1.d) includes cluttered background which makes HOWs less distinguishable.

Part-based models [2] are two-layer structures where the root node represent the entire object while the nodes at the second layers correspond to the parts. Unlike spatial pyramids, the nodes are allowed to move to account for large deformations of object parts. But part-based models also have two limitations. Firstly, the appearance models of the parts, which is based on HOGs [2], is not suitable for regions with rich texture properties where gradients are not very informative. Secondly, the shallow structure (i.e., lack of a third layer) limits the representation of detailed appearance of the object and prevents the representation of small scale shape deformations.

This paper presents a new representation, called "Active Mask Hierarchies (AMH)", which offers a richer way to represent appearance variations and shape deformations. Our approach combines spatial pyramids and part-based models into a single representation. First observe that an active mask hierarchy can be considered as a spatial pyramid with relaxed bonds – hence "active" (see fig. 1.(b)). It can be represented by a tree structure where nodes at consecutive layers are vertically related, and assigned latent position variables to encode displacements of parts. Similarly active mask hierarchies can be thought of as a three-layer part-based model where "parts" together with their connections are simply designed as the active cells at different layers which are organized in a form of multi-level grids. As a result, the complicated procedure of part selection [2] is avoided. We will show that the multi-level grid design does not prevent us from achieving good performance.

Cells at different levels of the active mask hierarchy have appearance features based on HOGs and HOWs so as to model both structured and textured regions. Moreover, we assign a dictionary of masks with various binary shape patterns (fig. 1.(c)) to all nodes which enable the part to deal with variations in the shape (i.e., overcome the restriction to regular rectangular templates). The features are only measured in the white areas specified by the masks. For example, masks (fig. 1.(d) and 1.(e)) at the top layer give the coarse descriptions of the boundary of the entire object. The active masks at the lower layers (fig.1.(f)) with displacements combine to represent the object parts more accurately. The selection of masks is performed by weighting their importance.

Learning the hierarchical model is a latent structural SVM problem [6] which can be solved by the concave-convex procedure (CCCP). CCCP has been successfully applied to learning models for object detection [7,8]. In order to reduce the training cost we use the variant called incremental concave-convex procedure (iCCCP) first reported in [8]. iCCCP allows us to learn hierarchical models using a large-scale training set efficiently.

Our experimental results demonstrate that the active mask hierarchies achieve state-of-the-art performance evaluated on the challenging public PASCAL 2007 and 2009 datasets [9, 10]. As we show, the proposed method performs well at detecting both structured objects and textured objects.

2 Related Work

Hierarchical decomposition has also been explored in object recognition and image segmentation, such as [11, 12, 13]. Our use of shape masks is partially inspired by Levin and Weiss's fragments [14], Torralba et al. 's spatial mask [15] and by Zhu et al.'s recursive segmentation templates [16]. But [14,16] are applied to segmentation and not to object detection. The masks used in [15] are not associated with latent postion variable. The idea of "active" parts is similar in spirit to Wu et al.'s active basis model [17], which does not involve a hierarchy. Schnitzspan et al.'s [18] uses a hierarchical models, but does not contain the shape masks.

There has been much related work on object detection, including [1,2,8,19,20]. [1,2,8] focus on the modeling of objects. Vedaldi et al. [1] present multiple kernel learning applied to spatial pyramids of histograms of features. They use a cascade of models and non-linear RBF kernels. Felzenszwalb et al. [2] propose latent SVM learning for part-based models and explore the benefit of post-processing (eg, incorporating contextual information). As we will show in the experiments, our system gives better performance without needing these "extras". We use the iCCCP learning method developed in [8], but [8] does not use shape masks or HOWs (which give significant performance improvement as reported in the experimental section).

Instead of improving the representation of objects, both [19] and [20] focus on using global contextual cues to improve the performance of object detection. Desai et al. [19] make use a set of models from different object categories. [20] considers global image recognition and local object detection jointly.

3 Active Mask Hierarchies

In this section, we first formulate object learning as a latent structural SVM learning problem and then describe the active mask hierarchy representation. Finally, we briefly smmarize the optimization method for training and the inference algorithm for detection.

3.1 Active Mask Hierarchies and Latent Structural SVM

The goal of the AMH model is to detect whether an object with class label y is present in an image region x. The AMH model has latent variables $h = (V, \boldsymbol{p})$ (i.e. not specified in the training set), where V labels the mixture component and $\boldsymbol{p}$ specifies the positions of the object masks.

The AMH is specified by a function $w \cdot \Phi(x, y, h)$ where w is a vector of parameter weights (to be learnt) and Φ is a feature vector. Φ has two types of terms: (i) appearance terms $\Phi_A(x, y, h)$ which relate features of the image x to object classes y, components V, and mask positions $\boldsymbol{p}$; (ii) shape terms $\Phi_S(y, h)$ which specify the relationships between the positions of different masks and which are independent of the image x.

The *inference task* is to estimate the class label y and the latent states h by maximizing the discriminant function (assuming w is known):

$$F_w(x) = \underset{y,h}{\operatorname{argmax}}[w \cdot \Phi(x, y, h)] \tag{1}$$

The *learning task* is to estimate the optimal parameters w from a set of training data (x_1, y_1, h_1),...,(x_N, y_N, h_N). We formulate the learning task as latent structural SVM learning. The object labels $\{y_i\}$ of the image regions $\{x_i\}$ are known but the latent variables $\{h_i\}$ are unknown (recall that the latent variables encode the mask positions $\boldsymbol{p}$ and the model component V). The task is to find the weights w which minimize an objective function $J(w)$:

$$J(w) = \frac{1}{2}||w||^2 + C \sum_{i=1}^{N} \left[\max_{y,h}[w \cdot \Phi_{i,y,h} + L_{i,y,h}] - \max_{h}[w \cdot \Phi_{i,y_i,h}] \right] \quad (2)$$

where C is a fixed number, $\Phi_{i,y,h} = \Phi(x_i, y, h)$ and $L_{i,y,h} = L(y_i, y, h)$ is a loss function. For our object detection problem $L(y_i, y, h) = 1$, if $y_i = y$, and $L(y_i, y, h) = 0$ if $y_i \neq y$ (note $L(.)$ is independent of the latent variable h).

Solving the optimization problem in equation (2) is difficult because the objective function $J(w)$ is non-convex (because the fourth term $-\max_h[w \cdot \Phi_{i,y_i,h}]$ is a concave function of w). Following Yu and Joachims [6] we use the concave-convex procedure (CCCP) [21] which is guaranteed to converge at least to a local optimum. We note that CCCP has already been applied to learning models for object detection [7, 8]. We briefly describe CCCP and its application to latent SVMs in section 3.3.

In practice, the inner product in the discriminative function in equation (1) can be expressed as a summation of kernel functions [1]:

$$w \cdot \Phi(x, y, h) = \sum_{i,y',h'} \alpha_{i,y',h'} \mathcal{K}(\Phi_{i,y',h'}, \Phi_{x,y,h}) \quad (3)$$

where $\alpha_{i,y',h'}$ are weights for support vectors obtained by solving equation (2) and $\mathcal{K}(\Phi_{i,y',h'}, \Phi_{x,y,h})$ is a positive definite kernel, which can be represented by a linear (convex) combination of kernels:

$$\mathcal{K}(\Phi_{i,y',h'}, \Phi_{x,y,h}) = \sum_{k} d_k \mathcal{K}_k(\Phi_{i,y',h'}, \Phi_{x,y,h}) \quad (4)$$

where $\mathcal{K}_k(\Phi_{i,y',h'}, \Phi_{x,y,h})$ correspond to the appearance and shape kernels and d_k are their weights. We will introduce these kernels in section (3.2).

3.2 The Representation: Hierarchical Model and Feature Kernels

An AMH represents an object class by a mixture of two 3-layer tree-structured models. The structure of the model is shown in fig. 1.(b). The structure used in our experiments is slightly different, but, for the sake of simplicity, we will use this structure to illustrate the basic idea and describe the difference in section 4.5 .

The first layer has one root node which represents the entire object. The root node has four child nodes at the second layer in a 2×2 grid layout where each cell represents one fourth of an object. Each node at the second layer has 4 child nodes at the third layer which contains 16 nodes in a 4×4 grid layout. There are 21 $(1 + 2 \times 2 + 4 \times 4)$ nodes in total. Note that the cells in the spatial pyramid (figure 1.(a)) are not connected.

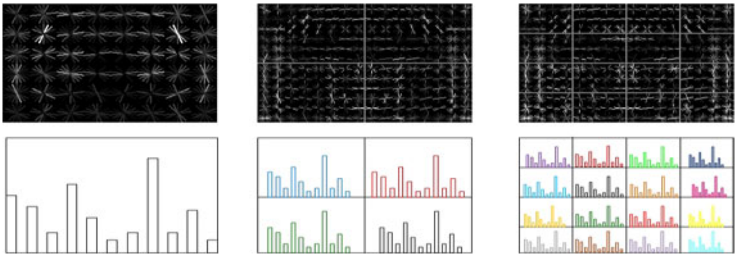


Fig. 2. The top three panels show the Histogram of Oriented Gradients (HOGs). The bottom three panels show the Histogram Of Words (HOWs) extracted within different cells. The visual words are formed by using SIFT descriptors. The columns from left to right correspond to the top to bottom levels of the active hierarchy.

The numbers of layers and nodes are the same for different object classes and mixture components. But their aspect ratios may be different. Each tree model is associated with latent variables $h = (V, \boldsymbol{p})$. $V \in \{1, 2\}$ is the index of the mixture components and $\boldsymbol{p} = ((u_1, v_1), (u_2, v_2), ..., (u_{21}, v_{21}))$ encodes the positions of all nodes. For an object class, let $y = +1$ denote object and $y = -1$ denote non-object. Let $a \in \{1, ..., 21\}$ index the nodes. $b \in Ch(a)$ indexes the child nodes of node a.

The feature vector for each mixture component V is defined as follows:

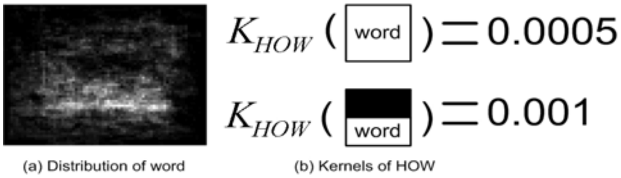
$$\Phi(x, y, \boldsymbol{p}) = \begin{cases} (\Phi_A(x, \boldsymbol{p}), \Phi_S(\boldsymbol{p})) & \text{if } y = +1 \\ 0 & \text{if } y = -1 \end{cases} \tag{5}$$

where $\Phi_A(x, \boldsymbol{p})$ is a concatenation of appearance feature vectors $\Phi_A(x, \boldsymbol{p}_a)$ which describe the image property of the corresponding regions specified by $\boldsymbol{p}_a$. $\Phi_S(\boldsymbol{p})$ is a concatenation of shape feature vectors $\Phi_S(\boldsymbol{p}_a, \boldsymbol{p}_b)$ which encode the parent-child spatial relationship of the nodes $(\boldsymbol{p}_a, \boldsymbol{p}_b)$. Note for different components V, we maintain separate feature vectors.

The appearance features consist of two types of descriptors (see fig. 2): (i) Histograms of Oriented Gradients (HOGs) $\Phi_{HOG}(x, \boldsymbol{p})$ [5] and (ii) Histograms of Words (HOWs) $\Phi_{HOW}(x, \boldsymbol{p})$ [4] extracted from SIFT descriptors [22] which are densely sampled. These two descriptors are complementary to each other for appearance representation. HOGs are suitable for structured regions where the image patches with specific oriented gradients (like car wheels, cat eyes, etc.) are located at certain position. On the other hand, HOW's advantages specialize at the textured regions where the small image patches encoded by visual words (like texton patches in cat body) appear randomly in a spatial domain. We followed the implementations of [2] to calculate the HOG descriptors, and [1] for the SIFT descriptors. Visual words are extracted by K-means using SIFT descriptors.

The HOW features are a vector of features calculated within the valid regions specified by the 11 shape masks, i.e.,

$$\Phi_{HOW}(x, \boldsymbol{p}) =< \Phi^1_{HOW}(x, \boldsymbol{p}), \Phi^2_{HOW}(x, \boldsymbol{p})..., \Phi^{11}_{HOW}(x, \boldsymbol{p}) > \tag{6}$$



(a) Distribution of word

(b) Kernels of HOW

Fig. 3. We illustrate the role of the mask and spatial variability ("active") of the most important HOW feature at the top level of the AMH. Figure (a) plots the maximum response (for all masks) of visual word over the horse dataset. Observe that the response is peaked but has big spatial variability so that the AMH can adapt to spatial position and deformation of the objects. Figure (b), the most successful mask is the horizontal bar – mask 5, see figure (1.c) – which has, for example, twice as high kernel values as mask 1 (the regular rectangle).

The shape masks associated with node a are located by the latent position variable p_a. They are forms of varied binary shape patterns (fig. 1.(c)) which encode large shape variations of object and parts in a coarse-to-fine manner. The regions activated for the feature calculations are the white areas specified by the masks. For instance, the masks (fig. 1.(d) and 1.(e)) at the top layer give the coarse descriptions of the boundary of the entire object. The active masks at the lower layers (fig. 1.(f)) with displacements combine to represent the object parts more accurately. The patterns of shape masks are designed so that the histograms of words within the masks can be calculated efficiently using integral image.

$\Phi_S(h)$ is a concatenation of shape features $\Phi_S(\boldsymbol{p}_a, \boldsymbol{p}_b), \forall a, b \in Ch(a)$, which encode the parent-child pairwise spacial relationship. More precisely, the shape features for a parent-child pair (a, b) are defined as $\Phi_S(\boldsymbol{p}_a, \boldsymbol{p}_b) = (\Delta u, \Delta v, \Delta u^2, \Delta v^2)$ where $(\Delta u, \Delta v)$ is the displacement of node b relative to its reference position which is determined by the position of the parent node a. Our 3-layer model has 80 $(4 \times 4 + 4 \times 16)$ shape features in total.

Now we have complete descriptions of the appearance and shape features. The kernel in equation (4) which combines the appearance and shape kernels is given by (note we only consider the nontrivial case, i.e., $y = +1$):

$$\mathcal{K}(\Phi_{i,y',h'}, \Phi_{x,y,h}) = \mathcal{K}_A(\Phi(x_i, \boldsymbol{p}'), \Phi(x, \boldsymbol{p})) + \mathcal{K}_S(\Phi(\boldsymbol{p}'), \Phi(\boldsymbol{p})) \qquad (7)$$

where $\mathcal{K}_S(\Phi(\boldsymbol{p}'), \Phi(\boldsymbol{p}))$ is the shape kernel which is a simple linear kernel, i.e. $\mathcal{K}_S(.,.) =< \Phi(\boldsymbol{p}'), \Phi(\boldsymbol{p}) >$. $\mathcal{K}_A(\Phi(x_i, \boldsymbol{p}'), \Phi(x, \boldsymbol{p}))$ is the appearance kernel which is given by the weighted sum of two types of appearance kernels:

$$d_1\mathcal{K}_1(\Phi_{HOG}(x_i, \boldsymbol{p}'), \Phi_{HOG}(x, \boldsymbol{p})) + d_2\mathcal{K}_2(\Phi_{HOW}(x_i, \boldsymbol{p}'), \Phi_{HOW}(x, \boldsymbol{p})) \qquad (8)$$

where d_1, d_2 are weights for two appearance kernels respectively. $\mathcal{K}_1(.,.)$ is a simple linear kernel, i.e., $\mathcal{K}_1(.,.) =< \Phi_{HOG}(x_i, p'), \Phi_{HOG}(x, \boldsymbol{p}) >$. $\mathcal{K}_2(.,.)$ is a quasi-linear kernel [1], i.e., $\mathcal{K}_2(.,.) = \frac{1}{2}(1 - \mathcal{X}^2(\Phi_{HOW}(x_i, \boldsymbol{p}'), \Phi_{HOW}(x, \boldsymbol{p})))$, which can be calculated efficiently using the technique proposed in [23]. Note that unlike [1], the non-linear RBF kernels are not used here.

Figure (3) shows how the appearance kernels of the HOWs and the shape masks work. Recall that each HOW is computed for 11 masks, and the positions of these masks vary depending on the input image. Firstly, we explore the spatial variation of the maximum response of the HOW feature (for all masks) for the horse dataset. Our results, see figure (3.a) show that the maximum response is spatially peaked in the lower center of the image window containing the object. But the position of the response varies considerably due to the variation in shape and location of the object. Secondly, by examining the mask kernel values, we see that mask 5 (horizontal bar) is the most effective when evaluated on this database and, see figure (3.b), has kernel value which is twice as high as mask 1 (regular rectangle).

The free parameter in equation (8) is the ratio r of two weights $d1 : d2$. In our experiments, the ratio r is selected by cross validation as explored in [4]. It is possible to improve the performance using more recent technique on feature combination [24]. We leave it as future work.

Now we have a complete description for the representation of active mask hierarchies.

3.3 Optimization by CCCP

Learning the parameters w of the AMH model requires solving the optimization problem specified in equation (2). Following Yu and Joachims [6], we express the objective function $J(w) = f(w) - g(w)$ where $f(.)$ and $g(.)$ are convex functions given by:

$$\begin{aligned} f(w) &= \left[\frac{1}{2}||w||^2 + C\sum_{i=1}^{N} \max_{y,h}[w \cdot \Phi_{i,y,h} + L_{i,y,h}]\right] \\ g(w) &= \left[C\sum_{i=1}^{N} \max_{h}[w \cdot \Phi_{i,y_i,h}]\right] \end{aligned} \tag{9}$$

The concave-convex procedure (CCCP) [21] is an iterative algorithm which converges to a local minimum of $J(w) = f(w) - g(w)$. When $f(\cdot)$ and $g(\cdot)$ take the forms specified by equation (9), then CCCP reduces to two steps [6] which estimate the latent variables and the model parameters in turn (analogous to the two steps of the EM algorithm):

Step (1): Estimate the latent variables h by the best estimates given the current values of the parameters w: $h^* = (V^*, \boldsymbol{p}^*)$ (this is performed by the inference algorithm described in the following section).

Step (2): Apply structural SVM learning to estimate the parameters w using the current estimates of the latent variables h:

$$\min_{w} \frac{1}{2}||w||^2 + C\sum_{i=1}^{N} \left[\max_{y,h}[w \cdot \Phi_{i,y,h} + L_{i,y,h}] - w \cdot \Phi_{i,y_i,h_i^*}\right] \tag{10}$$

We perform this structural SVM learning by the cutting plane method [25] to solve equation (10).

In this paper, we use a variant called *incremental CCCP* (iCCCP) first reported in [8]. The advantage of iCCCP is that it uses less training data and hence makes the learning more efficient. The kernel in equation (7) is applied without changing the training algorithm.

3.4 Detection: Dynamic Programming

The inference task is to estimate $F_w(x) = \text{argmax}_{y,h}[w \cdot \Phi(x, y, h)]$ as specified by equation (1). The parameters w and the input image region x are given. Inference is used both to detect objects after the parameters w have been learnt and also to estimate the latent variables during learning (Step 2 of CCCP).

The task is to estimate $(y^*, h^*) = \text{argmax}_{y,h}[w \cdot \Phi(x, y, h)]$. The main challenge is to perform inference over the mask positions $\boldsymbol{p}$ since the remaining variables y, V take only a small number of values. Our strategy is to estimate the $\boldsymbol{p}$ by dynamic programming for all possible states of V and for $y = +1$, and then take the maximum. From now on we fix y, V and concentrate on $\boldsymbol{p}$.

First, we obtain a set of values of the root node $\boldsymbol{p}_1 = (u_1, v_1)$ by exhaustive search over all subwindows at different scales of the pyramid. Next, for each location (u_1, v_1) of the root node we use dynamic programming to determine the best configuration $\boldsymbol{p}$ of the remaining 20 parts. To do this we use the recursive procedure:

$$F(x, \boldsymbol{p}_a) = \sum_{b \in Ch(a)} \max_{\boldsymbol{p}_b} \{F(x, \boldsymbol{p}_b) + w \cdot \Phi_S(\boldsymbol{p}_a, \boldsymbol{p}_b)\} + w \cdot \Phi_A(x, \boldsymbol{p}_a) \tag{11}$$

where $F(x, \boldsymbol{p}_a)$ is the max score of a subtree with root node a. The recursion terminates at the leaf nodes b where $F(x, \boldsymbol{p}_b) = \Phi_A(x, \boldsymbol{p}_b)$. This enables us to efficiently estimate the configurations $\boldsymbol{p}$ which maximize the discriminant function $F(x, \boldsymbol{p}_1) = \max_{\boldsymbol{p}} w \cdot \Phi(x, \boldsymbol{p})$ for each V and for $y = +1$.

The bounding box determined by the position (u_1, v_1) of the root node and the corresponding level of the image pyramid is output as an object detection if the score $F(x, \boldsymbol{p}_1) >$ is greater than certain threshold.

In our implementations, $w \cdot \Phi_A(x, \boldsymbol{p}_a)$ is replaced by the appearance kernel $\mathcal{K}_A(\Phi(x_i, \boldsymbol{p}'), \Phi(x, \boldsymbol{p}))$ described in equation (8).

4 Experiments

The PASCAL VOC 2007 [9] and 2009 [10] datasets were used for evaluation and comparison. The PASCAL 2007 is the last version for which test annotations are available. There are 20 object classes which consist of 10000 images for training and testing. We follow the experimental protocols and evaluation criteria used in the PASCAL Visual Object Category detection contest 2007. A detection is considered correct if the intersection of its bounding box with the groundtruth bounding box is greater than 50% of their union. We compute precision-recall (PR) curves and score the average precision (AP) across a test set.

Table 1. Comparisons of performance on the PASCAL 2007 dataset. The numbers are the average precisions per category obtained by different methods. "UoCTTI-1" and "UoCTTI-2" report the results from [2] with and without special post-processing, respectively. "MKL-1" and "MKL-2" show the results obtained by [1] using quasi-linear kernels and RBF kernels, respectively.

Methods	Active Mask Hierarchies	no mask [8]	UoCTTI-1 [2]	UoCTTI-2 [2]	MKL-1 [1]	MKL-2 [1]	[19]	[20]	[18]
comments	HOG+HOW	HOG	Part-Based	+ context	pyramid	+RBF			
Ave. Precision	.338	.296	.268	.298	.291	.321	.271	.289	.275

Table 2. Performance Comparisons on the 20 PASCAL 2007 categories [9]. "Active Mask Hierarchies" refers to the proposed method in this paper. "UoCTTI-1" and "UoCTTI-2" report the results from [2] with and without special post-processing, respectively. "MKL-1" and "MKL-2" show the results obtained by [1] using quasi-linear kernels and RBF kernels, respectively. "V07" is the best result for each category among all methods submitted to the VOC 2007 challenge. Our method outperforms the other methods in 11 categories. The average APs per category of all methods are shown in the second column which have the corresponding numbers in table (1).

class	Ave.	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow
Active Mask Hierarchies (AMH)	**.338**	.348	.544	**.155**	.146	.244	**.509**	**.540**	**.335**	**.206**	.228
Hierarchy without masks [8]	.296	.294	.558	.094	.143	**.286**	.440	.513	.213	.200	.193
UoCTTI-1 (Part-based) [2]	.268	.290	.546	.006	.134	.262	.394	.464	.161	.163	.165
UoCTTI-2 (Part-based) [2]	.298	.328	**.568**	.025	**.168**	.285	.397	.516	.213	.179	.185
MKL-1 (Pyramid-based) [1]	.292	.366	.425	.128	.145	.151	.464	.459	.255	.144	.304
MKL-2 (Pyramid-based) [1]	.321	**.376**	.478	.153	.153	.219	.507	.506	.300	.173	**.330**
V07 [9]	—	.262	.409	.098	.094	.214	.393	.432	.240	.128	.140

	Ave.	table	dog	horse	mbike	person	plant	sheep	sofa	train	tv
Active Mask Hierarchies	**.338**	**.344**	**.241**	**.556**	**.473**	.349	**.181**	.202	**.303**	.413	.433
Hierarchy without masks [8]	.296	.252	.125	.504	.384	.366	.151	.197	.251	.368	.393
UoCTTI-1 (Part-based) [2]	.268	.245	.050	.436	.378	.350	.088	.173	.216	.340	.390
UoCTTI-2 (Part-based) [2]	.298	.259	.088	.492	.412	**.368**	.146	.162	.244	.392	.391
MKL-1 (Pyramid-based) [1]	.292	.190	.160	.490	.460	.215	.110	**.245**	.264	.426	.408
MKL-2 (Pyramid-based) [1]	.321	.225	.215	.512	.455	.233	.124	.239	.285	**.453**	**.485**
V07 [9]	—	.098	.162	.335	.375	.221	.120	.175	.147	.334	.289

4.1 The Detection Results on the PASCAL Dataset

We compared our approach with other representative methods reported in the PASCAL VOC detection contest 2007 [9] and other more recent work [2, 1, 19, 20, 18]. Table (1) reports the Average Precisions per category (averaged over 20 categories) obtained by different methods. The comparisons in table (1) show that the active mask hierarchies (AMH) outperform other methods including state-of-the-art systems, i.e., [1] and [2].

It is important to realize that our result (0.338 AP) is obtained by a single model while all other methods' final results rely on combining multiple models. For instance, "MKL-2" [1] (0.321 AP) uses cascade of models where non-linear RBF kernels and more features are used. "UoCTTI-2" [2] (0.298 AP) combines the detections output by models of all categories to access contextual information. It is clear that the additional processing improves the performance. For

Table 3. Performance Comparisons on the 20 PASCAL 2009 categories [10]. The approaches in the first column are described in table (2).

class	Ave.	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow
Active Mask Hierarchies (AMH)	.293	.432	.404	.135	.141	.271	.407	.355	.330	.172	.187
UoCTTI-2 (Part-based) [2]	.279	.395	.468	.135	.150	.285	.438	.372	.207	.149	.228
MKL-2 (Pyramid-based) [1]	.277	.478	.398	.174	.158	.219	.429	.277	.305	.146	.206

	Ave.	table	dog	horse	mbike	person	plant	sheep	sofa	train	tv
Active Mask Hierarchies	.293	.227	.219	.371	.444	.398	.129	.207	.247	.434	.342
UoCTTI-2 (Part-based) [2]	.279	.087	.144	.380	.420	.415	.126	.242	.158	.439	.335
MKL-2 (Pyramid-based) [1]	.277	.223	.170	.346	.437	.216	.102	.251	.166	.463	.376

example, the model with RBF kernels [1] improves by 0.03 AP and the post-processing used in [2] contributes 0.03 AP.

To give a better understanding how significant the improvement made by AMH is, three other recent advances are listed for comparisons. All of them explore the combination of multiple models as well. They achieve 0.271 ([19]), 0.289 ([20]) and 0.275([18]). [19] makes use of multiple models of different categories. [20] considers the recognition and detection jointly. [18] seeks to rescore the detection hypotheses output by [2].

In table (3), we report the performance evaluated on the PASCAL 2009 dataset. It also shows that our method is comparable with “MKL-2” [1] and “UoCTTI-2” [2]. In summary, our system built on a single model outperforms other alternative methods. It is reasonable to expect that our method with additional processing (e.g., RBF kernels, contextual cues, etc.) as used in other methods will achieve even better performance.

4.2 Active Mask Hierarchies, Spatial Pyramid and Part-Based Model

As we discussed before, spatial pyramid and part-based model can be unified in the representation of active mask hierarchies (AMH). It is of interest to study how differently (or similarly) each method performs in specific classes which have different scales of shape deformation and appearance variations. We show the detailed comparisons of the results on 20 object classes (PASCAL2007) in table (2). Our method obtains the best AP score in 11 out of 20 categories while MKL-2 using RBF kernels achieves the best performance in 4 categories. In order to show the advantage of the representation of AMH, it is more appropriate to compare AMH with MKL-1 which uses the same quasi-linear kernel of spatial pyramid, and “UoCTTI-1” which uses a part-based model only. Note AMH outperforms “UoCTTI-1” by 0.07 AP , “MKL-1” by 0.05 AP and [8] by 0.04 AP. Therefore, the improvement made by AMH is significant.

4.3 Benefit of Shape Masks

Table (1) shows that the active mask hierarchies (AMH) with both HOGs and HOWs outperform [8] by 0.04 AP. The detailed comparisons on 20 object classes

(PASCAL 2007) are shown in table (2). Recall that [8] uses HOGs only, and does not contain the shape masks and the HOW features. We quantify the gain contributed by HOWs and the shape masks. Figure (4) shows the PR curves of the three models using HOGs only [8], AMH (HOGs+HOWs) attached with one shape mask (regular rectangle), and AMH (HOGs+HOWs) with a dictionary of shape masks, respectively. HOWs improve the performance for bus by 0.03AP and horse by 0.01AP, but degrade the performance for car by less than 0.01AP. Adding shape masks makes improvement by 0.07, 0.03, 0.05 APs, for bus, car and horse, respectively.

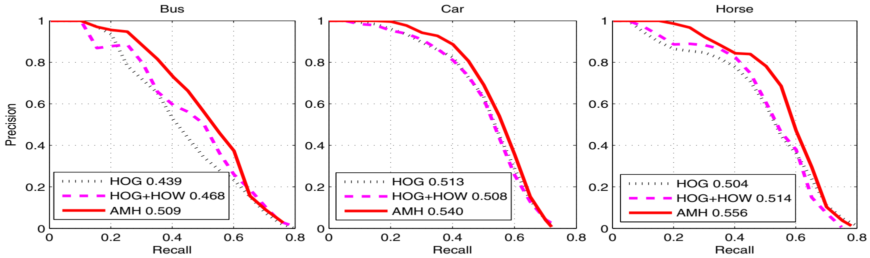


Fig. 4. The benefit of shape masks. "HOG" and "HOG+HOW" refer to the simple active hierarchy models without shape masks using HOGs only, and both HOGs and HOWs, respectively. "AMH" is the active shape hierarchy with both HOGs and HOWs. Three panels show the precision-recall curves evaluated on the bus, car, horse datasets.

4.4 Weights of HOGs and HOWs

The ratio of weights of appearance kernels for HOGs and HOWs is selected by cross validation. Three values of the ratio r, i.e., $d1 : d2 = 0.5, 1.0, 2.0$ are tested. Figure (5) shows the PR curves of the models obtained by the appearance kernels with three values of r. For car, the result is less sensitive for the ratio, but for motorbike and horse, the maximum differences of performance are about 0.07 AP and 0.10 AP, respectively. The training cost is affordable if only one parameter needs to be selected. If more parameters are used, [24] can be used to learn the combination of appearance features in an efficient way.

4.5 Implementation Details

All experiments are performed on a standard computer with a 3Ghz CPU. C is set to 0.005 for all classes. The detection time per image is 50 seconds. There are 300 visual words which are extracted by k-means where the color SIFT descriptors are used. The structure of the hierarchy used in our experiment is slightly different from the one shown in figure 1. In our implementations, the number of nodes at from top to bottom levels are $1(1 \times 1), 9(3 \times 3), 36(6 \times 6)$. The nodes are organized in regular multi-level grids. The HOW features Φ_{HOW} at different layers of the pyramid are associated with fixed weights, i.e., 6:2:1,

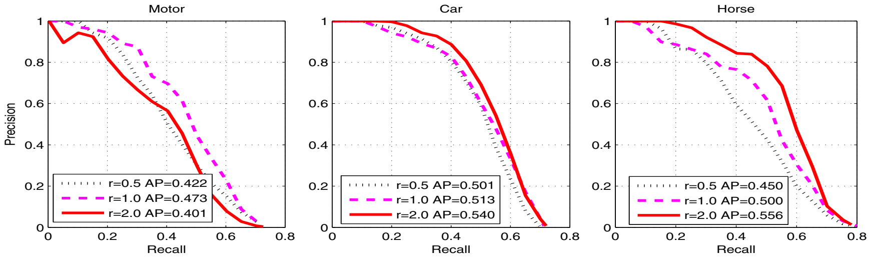


Fig. 5. We compare the performance of AMHs with different ratios of weights of HOGs and HOWs. Three panels plot the precision-recall curves for the bus, car and horse datasets.

for all categories. As suggested by [4], other settings might further improve the performance. The settings of all free parameters used in the PASCAL 2007 and 2009 datasets are identical.

5 Conclusion

This paper describes a new active mask hierarchy model for object detection. This active hierarchy enables us to encode large shape deformation of object parts explicitly. The dictionary of masks with varied shape patterns increases our ability to represent shape and appearance variations. The active mask hierarchy uses histograms of words (HOWs) and oriented gradients (HOGs) to give rich appearance models for structured and textured image regions. The resulting system outperforms spatial pyramid and part-based models, and comparable with the state-of-the-art methods by evaluation on the PASCAL datasets.

Acknowledgments. Funding for this work was provided by NGA NEGI-1582-04-0004, MURI Grant N00014-06-1-0734, AFOSR FA9550-08-1-0489, NSF IIS-0917141 and gifts from Microsoft, Google and Adobe. Thanks to the anonymous reviewers for helpful feedback. We thank Antonio Torralba and Jianxiong Xiao for discussions.

References

1. Vedaldi, A., Gulshan, V., Varma, M., Zisserman, A.: Multiple kernels for object detection. In: Proceedings of the International Conference on Computer Vision (2009)
2. Felzenszwalb, P.F., Grishick, R.B., McAllister, D., Ramanan, D.: Object detection with discriminatively trained part based models. PAMI (2009)
3. Lazebnik, S., Schmid, C., Ponce, J.: Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In: CVPR (2006)
4. Bosch, A., Zisserman, A., Munoz, X.: Representing shape with a spatial pyramid kernel. In: CIVR (2007)

5. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: CVPR, pp. 886–893 (2005)
6. Yu, C.N.J., Joachims, T.: Learning structural svms with latent variables. In: International Conference on Machine Learning (ICML) (2009)
7. Vedaldi, A., Zisserman, A.: Structured output regression for detection with partial occulsion. In: Proceedings of Advances in Neural Information Processing Systems (NIPS) (2009)
8. Zhu, L., Chen, Y., Yuille, A., Freeman, W.: Latent hierarchical structural learning for object detection. In: CVPR (2010)
9. Everingham, M., Van Gool, L., Williams, C.K.I., Winn, J., Zisserman, A.: The PASCAL Visual Object Classes Challenge, VOC 2007 Results (2007), `http://www.pascal-network.org/challenges/VOC/voc2007/workshop/index.html`
10. Everingham, M., Van Gool, L., Williams, C.K.I., Winn, J., Zisserman, A.: The PASCAL Visual Object Classes Challenge, VOC 2009 Results (2009), `http://www.pascal-network.org/challenges/VOC/voc2009/workshop/index.html`
11. Epshtein, B., Ullman, S.: Feature hierarchies for object classification. In: Proceedings of IEEE International Conference on Computer Vision, pp. 220–227 (2005)
12. Zhu, S., Mumford, D.: A stochastic grammar of images. Foundations and Trends in Computer Graphics and Vision 2, 259–362 (2006)
13. Storkey, A.J., Williams, C.K.I.: Image modelling with position-encoding dynamic. PAMI (2003)
14. Levin, A., Weiss, Y.: Learning to combine bottom-up and top-down segmentation. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) ECCV 2006. LNCS, vol. 3954, pp. 581–594. Springer, Heidelberg (2006)
15. Torralba, A., Murphy, K., Freeman, W.: Sharing visual features for multiclass and multiview object detection. PAMI (2007)
16. Zhu, L., Chen, Y., Lin, Y., Lin, C., Yuille, A.: Recursive segmentation and recognition templates for 2d parsing. In: Advances in Neural Information Processing Systems (2008)
17. Wu, Y.N., Si, Z., Fleming, C., Zhu, S.C.: Deformable template as active basis. In: ICCV (2007)
18. Schnitzspan, P., Fritz, M., Roth, S., Schiele, B.: Discriminative structure learning of hierarchical representations for object detection. In: Proc. CVPR (2009)
19. Desai, C., Ramanan, D., Fowlkes, C.: Discriminative models for multi-class object layout. In: Proceedings of the International Conference on Computer Vision (2009)
20. Harzallah, H., Jurie, F., Schmid, C.: Combining efficient object localization and image classification. In: ICCV (2009)
21. Yuille, A.L., Rangarajan, A.: The concave-convex procedure (cccp). In: NIPS, pp. 1033–1040 (2001)
22. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. International Journal of Computer Vision 60, 91–110 (2004)
23. Maji, S., Berg, A.C., Malik, J.: Classification using intersection kernel support vector machines is efficient. In: CVPR (2008)
24. Gehler, P., Nowozin, S.: On feature combination for multiclass object classification. In: ICCV (2009)
25. Tsochantaridis, I., Hofmann, T., Joachims, T., Altun, Y.: Support vector machine learning for interdependent and structured output spaces. In: Proceedings of International Conference on Machine Learning (2004)