Semantic Hierarchies for Recognizing Objects and Parts

Boris Epshtein Shimon Ullman
Department of Computer Science and Applied Mathematics
Weizmann Institute of Science
Rehovot, ISRAEL, 71600
{boris.epshtein , shimon.ullman}@weizmann.ac.il

Abstract

This paper describes the construction and use of a novel representation for the recognition of objects and their parts, the semantic hierarchy. Its advantages include improved classification performance, accurate detection and localization of object parts and sub-parts, and explicitly identifying the different appearances of each object part. The semantic hierarchy algorithm starts by constructing a minimal feature hierarchy and proceeds by adding semantically equivalent representatives to each node, using the entire hierarchy as a context for determining the identity and locations of added features. Part detection is obtained by a bottom-up top-down cycle. Unlike previous approaches, the semantic hierarchy learns to represent the set of possible appearances of object parts at all levels, and their statistical dependencies. The algorithm is fully automatic and is shown experimentally to substantially improve the recognition of objects and their parts.

1. Introduction

A number of recent studies have shown the benefits of using a hierarchical feature representation for recognition [1-5]. In this approach, top-level features are represented by collections of selected sub-features. For example, an eye can be represented as a set of smaller features, depicting a left eye corner, an eye pupil and a right eye corner. This feature organization provides better tolerance to learned geometric distortions and illumination changes compared with a non-hierarchical representation. It also makes it possible to recognize not only the top-level objects, but also all their parts and sub-parts at different levels.

A second extension which proved useful for recognizing object parts is the use of so-called semantic features [6], which are used to represent semantically equivalent object parts. The same part in an object class, and even a single object, can have multiple different appearances, such as different shapes of a car window, an open vs. closed mouth, etc. To obtain good recognition of objects and parts, the different possible appearances need to be learned and represented. For example, a semantic feature "eye" can be comprised of object fragments depicting closed eye, open eye, etc. The semantic features are used for representing different appearances of the same object part.

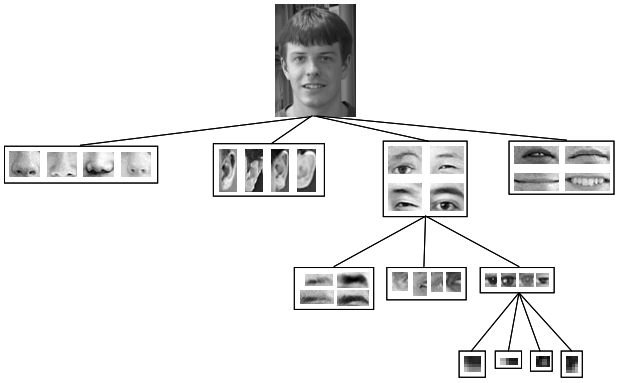

Figure 1. Schematic illustration of a semantic hierarchy. A face is represented as a combination of parts and sub-parts. Each part is represented as a semantic equivalence set of different possible appearances. The proposed scheme is the first to extract and use semantic parts in feature hierarchies.

In this paper, we develop a novel representation, called semantic hierarchy, which combines and extends the use of features hierarchy with the representation of semantic features at each hierarchy node. We describe an algorithm for the construction of complete semantic hierarchies from examples, and demonstrate its use for detecting objects as well as their parts. The proposed method overcomes limitations of previous algorithms, and efficiently learns useful semantic hierarchies from examples. A schematic illustration of a semantic hierarchy is shown in Figure 1.

As in previous approaches, an object and its parts are represented by a probabilistic model, which uses a learned probability distribution $p(C, \underline{X}, \underline{F})$ to infer the class C, and the parts $\underline{X}$ from the observed features $\underline{F}$. In the semantic hierarchy described below, we treat different appearances of the same part as different values of a given variable in the model. For example, a mouth and a nose are two parts,

1-4244-1180-7/07/$25.00 ©2007 IEEE

represented by two variables X_i, X_j in the representation. In contrast, open and closed mouths are represented by two possible values of the mouth variable.

We represent an object class (such as a car, or horse) by a probabilistic graphical model which has a tree structure, describing the class in terms of parts and sub-parts. A variable X_i in the model corresponds to an object part, and the possible values of X_i correspond to different appearances of the part at different locations. As we will see, this captures the statistical dependencies between parts and their different appearances.

The model construction algorithm proceeds along the following main stages. First, a minimal hierarchy of non-semantic features is constructed. Next, for each training image the optimal location of each feature is determined. For each missing feature, its optimal position is predicted, using the position of the parent feature and the learned parent-child geometric relationship. From the set of missing and nearly missing features, the best representatives are determined and added to the set of semantic appearances of each hierarchy node. Finally, a round of hierarchical decomposition is applied to the newly added appearances, to extract sub-parts that are specific for them. During recognition, the object together with its parts and sub-parts are detected by a bottom-up, top-down computation applied to the hierarchy.

The rest of the paper is organized as follows. The next section reviews related previous approaches to feature hierarchies and semantically equivalent features. In Section 3 we describe the proposed algorithms for feature selection and classification. In Section 4 we show experimental results, comparing the proposed approach with previous hierarchical schemes. We conclude with general discussion of the results in Section 5.

2. Prior works

The hierarchical organization of features was studied recently in several schemes [1-5]. In all these algorithms, complex features are detected by combining the responses of simpler features. The hierarchies can be constructed in a bottom-up or top-down manner. The bottom-up approach [1,2,4,5] starts with low-level features and then groups them together into more complex entities. The top-down approach [3] starts with extracting top-level features, informative for the classification task, and successively breaks them down into smaller sub-parts. In all these schemes, the features are represented by single appearances. We demonstrate experimentally that our approach produces better recognition and part detection results.

Semantic features were used recently for representing different appearances of a given part [6]. The algorithm proposed in this work for discovering semantically equivalent features is inapplicable to our task for several reasons. First, extracting a separate context set for every part and sub-part in a hierarchy is computationally inefficient. Second, the method is not applicable to the lower level of the hierarchy, since for simple features the method will fail to find context fragments with stable geometric relationships. Our semantic hierarchy method uses the rest of the hierarchy as a context for a part being considered; consequently, the algorithm successfully finds the semantic appearances of visually ambiguous parts.

3. The semantic hierarchy

In this section, we first describe the probabilistic graphical model of the semantic hierarchy (3.1), and how it is used for classification and parts detection (3.2). We then describe how the semantic hierarchy is constructed (3.3), and how the model is trained by examples (3.4).

3.1. The probabilistic graphical model

The graphical model of the semantic hierarchy is a class-conditional mixture model: the distributions *p(Evidence|Class)* and *p(Evidence|Nonclass)* are modeled separately. The conditional distribution *p(Evidence|Class)* is modeled as a hierarchy of features and sub-features (Fig. 2a), and the conditional distribution *p(Evidence|Nonclass)* is modeled as a Naive Bayes model (Fig. 2b).

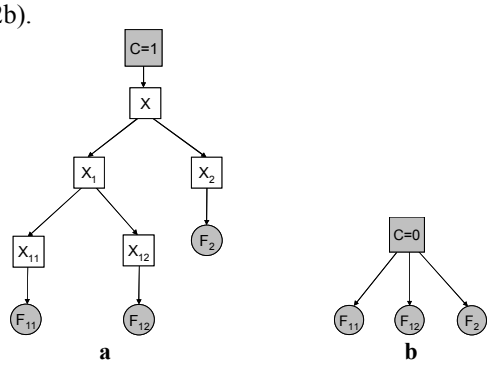

Figure 2. (**a**) Class model (**b**) Non-class model. F_i are the observable features, X is the entire object, X_i are object parts, and C is the class node. During recognition, the features F_i are observed in the images, and the computation infers the most likely values of X, X_i.

The hidden nodes X_i correspond to object parts and each node contains two discrete variables, $A(X_i)$ and $L(X_i)$. $A(X_i)$ is the number of a specific appearance of the object part. For example, if X_i corresponds to a mouth, $A(X_i)$=1 could signal the presence of a closed mouth in the image, $A(X_i)$=2 an open mouth. The variable $L(X_i)$ corresponds to the location of the part in the image, and can assume values in the range $[0...N]$, where N is the number of the image sites. The value $L(X_i)=0$ indicates that the part is not present in the image; values between 1 and N indicate that the part is located at the corresponding image site.

The hidden node X corresponds to the entire object and also contains a pair of variables, $A(X)$ and $L(X)$. The only difference compared with nodes X_i is that $L(X)$ assumes values in the range $[1..N]$, since for the class model the object is assumed to be present in the image.

The bottom-level observed nodes F_i correspond to the features observed in the image. Each such node contains $N*K$ values $S^1_1,...,S^1_N,...,S^K_1,...,S^K_N$, where N is again the number of image sites and K is the number of semantic appearances of the node X_i – the parent of F_i. Each value S^k_t represents image similarity between the feature represented by the k-th semantic appearance of X_i and the image patch centered at position t. The model used normalized cross-correlations, but other similarity measures (such as SIFT, or affine invariant features) can also be used. Note that only the "atomic" nodes X_i (corresponding to parts that are not broken further into sub-parts) have the observed feature nodes F_i attached to them. The full task we wish to accomplish (called "interpretation") is inferring the most likely values of X and all the parts X_i from the set of all observed features $\underline{F}$. This can be expressed as finding values for $\underline{X}$ (all the hidden variables) to maximize the probability $p(\underline{X}|\underline{F})$.

According to the class model, the probability $p(\underline{X}, \underline{F})$ factorizes into local terms:

$$p(\underline{X},\underline{F}) = p(X)\prod p(X_i \mid X_i^-)p(F_k \mid X_k) \qquad (1)$$

This decomposition assumes that each part X_i is conditionally independent of its non-descendants, given its parent X_i^-. Intuitively, this is a 'local context' assumption, namely, that the probability of a part X_i is given by its sub-parts and by its parent part in the hierarchy. This 'local context' property often holds for object images; for example, the detection of an eyebrow within a local eye region is relatively independent of more remote parts such as ears or chin. The model can also be extended to deal with additional interactions between parts when this assumption does not hold, but such extensions will not be discussed here further.

The parameters needed for the model are the probability of each node given its parent, as well as the prior on the uppermost node X. The model parameters can be divided into three types, as explained below.

p(A(X)=a, L(X)=l)

This is the probability of the object to be found in the class image in a location l and with the appearance a. In this work, this probability is assumed to be uniform.

p(F_i|A(X_i)=a, L(X_i)=l)

This is the probability of the feature F_i being observed, given that its parent node X_i has the appearance a and location l. Since F_i is a matrix of values $S^1_1,...,S^K_N$, corresponding to image similarities to different appearances of X_i in different locations, we can write, assuming conditional independence:

$$\begin{aligned} &p(F_i \mid A(X_i)=a, L(X_i)=l) = \\ &= p(S^1_1,...,S^K_N \mid A(X_i)=a, L(X_i)=l) = \\ &= \prod_{\substack{k=1...K \\ n=1...N}}^{N} p(S^k_n \mid A(X_i)=a, L(X_i)=l) \end{aligned} \qquad (2)$$

This probability can be further simplified as follows. We assume that given that the node X_i has the appearance a and location l, the similarity value S^a_l is distributed according to the "hit" distribution $p_h(S^a)$ (that is, depends only on the appearance a and is independent of the specific location l), and all other image similarities S^a_t for $t \neq l$ are distributed according to the "miss" distribution $p_m(S^a)$, representing the probability of observing similarity value S^a in an image site where the corresponding object part is missing. We found empirically that for image sites placed on a 4x4 pixel grid this approximation was sufficient, more accurate modeling of position effects did not produce better results. For the similarities between the image and all other appearances of X_i except a, we assume that all of them come from their respective "miss" distributions. This assumption holds in practice since the different semantic appearances of a node are selected to be substantially different visually. For each part X_i, the two distributions $p_h(S^k)$ and $p_m(S^k)$ are estimated separately for each appearance k at the training stage as described in Section 3.4. Combining this with (2), we get the decomposition:

$$p(S^1_1,...,S^K_N \mid A(X_i)=a, L(X_i)=l) = p_h(S^a_l)\prod_{\substack{k\neq a \\ n\neq l}} p_m(S^k_n) \qquad (3)$$

for the case $L(X_i) \neq 0$. Similarly,

$$p(S^1_1,...,S^K_N \mid L(X_i)=0) = \prod_{\substack{k=1...K \\ n=1...N}} p_m(S^k_n) \qquad (4)$$

for the case $L(X_i)=0$ (in this case, all the similarities are distributed according to their "miss" distributions). Finally, we divide all the probabilities $p(F_i|X_i)$ by the constant given by (4) to obtain an unnormalized score, proportional to the probability:

$$\begin{aligned} p(F_i \mid A(X_i)=a, L(X_i)=l) &\propto \frac{p(S^1_1,...,S^K_N \mid A(X_i)=a, L(X_i)=m)}{p(S^1_1,...,S^K_N \mid L(X_i)=0)} = \\ &= \frac{p_h(S^a_l)}{p_m(S^a_l)} \end{aligned} \qquad (5)$$

Division by a constant does not change any decision regarding classification or maximally probable state, but is convenient in the computation. In practice, this score is fitted by a piecewise linear function, using several histogram bins for which both numerator and denominator in (5) are non-zero.

p(A(X$_i$),L(X$_i$)|A(X$_i^-$),L(X$_i^-$))

This is the probability of an object part X_i given its parent, X_i^-. Since each hidden node contains two variables, appearance type and location, we make the following independence assumption:

$$p(A(X_i), L(X_i) \mid A(X_i^-), L(X_i^-)) = \\ = p(A(X_i) \mid A(X_i^-))\, p(L(X_i) \mid L(X_i^-)) \quad (6)$$

Here we assume that all the semantically equivalent parts share the same geometric relation with their parent. This assumption can be relaxed, but it often holds in practice. The second assumption is that the conditional probability can be factorized into the terms for appearance and location. The first term in (6) is the probability of a specific appearance of the child X_i given the appearance of its father, X_i^-. It is estimated at the training stage (Section 3.4). When a large number of appearances is used, there might be not enough training samples to estimate this probability robustly; in this case, we use Bayesian parameter estimation with a Dirichlet smoothing prior [12].

The second term in (6) is geometric probability. If X_i^- is not present ($L(X_i^-)=0$), the probability of observing a node X_i is assumed to be uniform, set to $\delta_0 /(N*K)$ for each position and appearance; consequently, the probability of $X_i=0$ is $1-\delta_0$. When X_i^- is present, the probability of $L(X_i)=0$ is set to $1-\delta_1$. The probability of X_i being present at some position in this case should sum up to δ_1. The parameters δ_0 ,δ_1 as all other parameters are learned from examples as described in Section 3.4. We make in this case the natural assumption that $p(L(X_i)|L(X_i^-))$ depends only on the coordinate differences between X_i and X_i^- rather than their absolute positions. We assumed that $p(L(X_i)|L(X_i^-))$ has a Gaussian distribution, with parameters estimated during learning. Alternatively, the distribution can be learned using the so-called 2D Parzen window estimation.

3.2. Classification and parts interpretation

Classification and interpretation are obtained by a bottom-up followed by a top-down stage. The first part is a bottom-up pass from the low-level features to the top-level node representing the class C. Formally, this pass computes the probability $p(\underline{F}|C=1)$. Since the non-class model is a Naïve Bayes model, the probability $p(\underline{F}|C=0)$ is given by multiplying the expressions (4) for all the leaf parts X_i. When these two values are known, the optimal classification decision is obtained by the likelihood ratio test:

$$\frac{p(C=1 \mid \underline{F})}{p(C=0 \mid \underline{F})} \propto \frac{p(\underline{F} \mid C=1)}{p(\underline{F} \mid C=0)} \quad (7)$$

In order to compute $p(\underline{F}|C=1)$ we proceed along the hierarchy tree in a bottom-up fashion, at each node X_i computing the set of probabilities of the evidence in the subtree under this node given the value of the node:

$$p(F(X_i) \mid X_i = k) = \\ = \prod_{X_i^+} \left(\sum_t p(F(X_{ij}) \mid X_{ij} = t)\, p(X_{ij} = t \mid X_i = k)\right) \quad (8)$$

here $F(X_i)$ is the evidence in the subtree under node X, and X_{ij} is the j-th child of the node X_i. The summation here is over all the locations and all the appearances of X_{ij}.The multiplication is over all the children of X_i (termed here X_i^+).

The most probable assignment for the variables:

$$D(\underline{X}) = \arg\max_{\underline{X}}\ p(\underline{X}, \underline{F} \mid C=1) \quad (9)$$

is computed by a top-down pass, using the standard max-product message passing algorithm for computing the MAP configuration [7]. A similar approach for computing the parts assignments is Hierarchical Maximal Marginal Probability (HMMP). First, the value of the object node X that maximizes $p(X|F(X))$ is determined and the assignment of X is fixed. The computation then proceeds down the tree, and the assingment for each node X_i that maximizes $p(F(X_i)|X_i)p(X_i|X_i^-)$ is computed. We used HMMP since it does not require running max-product computations, and uses only the results computed by the sum-product algorithm at the bottom-up step, thus increasing computational efficiency. Experimentally, the HMMP and MAP computations produced similar part assignments (HMMP result slightly superior, as verified by human observers).

3.3. The hierarchy construction

The first stage of constructing a semantic hierarchy is the construction of a simple hierarchy; that is, without the use of semantic features. This can be obtained using previous methods; we used the hierarchy of fragments and sub-fragments as in [3]. Briefly, the method selects a set of first-level informative fragments from a set of class and non-class training images. Each first-level fragment is then subdivided recursively into sets of informative sub-fragments, until the subdivision brings no gain in delivered information.

When the simple hierarchy is constructed, the main next stage is to identify and add semantically equivalent appearances to each hierarchy node. We use a set T of training images. For each training image T_n we compute the assignment for all the hidden variables, $H(\underline{X})$, using the HMMP procedure, as described in Section 3.2. A part X_i is considered missing in image T_n if $L(X_i)=0$ in $H(\underline{X})$. A part X_i is considered nearly missing, if it is detected with low confidence, that is:

$$\frac{p(F(X_i) \mid X_i = H(X_i))}{p(F(X_i) \mid L(X_i) = 0)} < \theta \quad (10)$$

here $H(X_i)$ is the assignment of X_i in $H(\underline{X})$, and the threshold θ is set automatically to the value producing the equal error rate in the recognition of the part X_i.

If a feature X_i is missing or nearly missing in T_n, but its parent X_i^- is present, the optimal location of X_i is determined using the location of the parent feature and the learned parent-child geometric relationship. The location of X_i is set to:

$$L(X_i) = \arg\max p(L(X_i) \mid L(X_i^-)) \quad (11)$$

The image patch representing the appearance of X_i in T_n is extracted at the predicted location with size equal to the size of the fragment originally encoded by X_i.

The above procedure is repeated for all training images; thereby, for each feature X_i, a set $M(X_i)$ of its missing and nearly missing appearances is collected. Next, for each feature, the set $M(X_i)$ is filtered: fragments that have less than k visually similar fragments in T are removed from the set. In the experiments, the value of k was set to 5. Next, the best representative appearances that maximize the classification performance on the training set are determined by a simple greedy algorithm as in [6] and these appearances are added to the set of semantic appearances of each hierarchy node. This procedure can identify semantically equivalent object parts regardless of their visual similarity (as measured by correlation, SIFT, etc.).

Finally, the newly added appearances at each node are also decomposed hierarchically. Since some sub-features can be shared between different appearances of the same part, we add sub-fragments that bring maximal increase of the mutual information to the already selected set of sub-fragments. Examples of feature hierarchies can be seen in Figures 6, 7.

3.4. Training the model

During training, example images are used to determine the parameters of the hierarchical network which represents the class model. The parameters of the model are $p(X)$, and the pair-wise probabilities $p(X_i|X_i^-)$, $p(F_k|X_k)$. The goal of the training stage is to derive these parameters from the training examples.

The model was trained using a variant of the EM algorithm [8], called Hard-EM [9]. The computation proceeds in an iterative manner, starting from an initial setting, and adjusting the parameters at the following stages, running over the set of training images.

The initial setting of the model parameters were as follows: the mean coordinate differences between a fragment and its parent fragment was set to their coordinate difference in the image from which they were extracted; the covariances were set to half the size of the parent fragment; and the probabilities of not detecting a child given that the parent is present, as well as the probability of detecting a child when the parent is not present were set to a fixed small value (0.001). The probabilities of semantic appearances $p(A(X_i)|A(X_i^-))$ were set by counting the percentages of detecting the corresponding fragments in the training data. Following the initial setting, at each iteration of Hard-EM the optimal configuration of parts was found by the HMMP procedure in Section 3.2. For each part X_i, at each training image the coordinate difference between its detected location $H(X_i)$ and the location of its parent feature $H(X_i^-)$ was computed (when both were detected in the image). A 2D Gaussian distribution was then fitted to match the empirical statistic. The percentage of cases when a part or its parent were not detected was counted, to estimate the parameters δ_0, δ_1. The probabilities $p(A(X_i)|A(X_i^-))$ of semantic compatibility between a part and its parent were also computed from the empirical counts.

4. Experimental results

This section describes the results of the empirical comparison of the semantic hierarchy with previous methods. As shown experimentally, the semantic hierarchy improves top-level classification as well as the detection and localization of object parts, especially parts with highly varying appearance. Section 4.1 shows comparison results for the task of object recognition. Section 4.2 shows comparison results for the task of parts detection, between simple and semantic hierarchies (comparisons between bottom-up and top-down detection will be described elsewhere). It shows improved parts detection and demonstrates the ability of the scheme to discover the different appearances of object parts.

4.1. Object recognition

The algorithm was applied to three object class datasets. The first was the Motorbikes set from the Caltech dataset [10], (400 motorbike, 269 background images in the training set, 400 motorbike, 227 background images in the test set, all images were grayscale, 224x148 pixels in size). The second dataset was a highly variable set of side images of horses, collected over the Web (222 horse images and 600 background images in the training set, 100 horse images and 1400 background images in the test set, all images were grayscale, 100x86 pixels in size). The third dataset contained 175 images of cars and 500 background images in the training set, and 175 car and 1500 background images in the test set, 200x150 pixels in size, collected over the Web. Examples of the object images are shown in Figure 3.

Semantic hierarchies were constructed for each object class and trained as described in Section 3. For

comparisons, a simple hierarchy (without semantically equivalent sets in each node) was built, using the same algorithm, except that the stage of extracting semantically equivalent features was not applied. The complete ROC curves were obtained by varying the threshold in the likelihood ratio test (7). For all classes, the semantic hierarchy produced significantly better results on all points of the ROC curve up to 99% of Hits ($p < 0.001$, t-test, 15 runs), as shown in Figure 4. To the best of our knowledge, the performance of semantic hierarchy on the Motorbikes dataset is comparable to the state of the art.

4.2. Parts detection

For this test, we compared the performance of the semantic hierarchy with previous methods on the difficult task of detecting and localizing object parts at multiple levels. The detection error rates were computed for several object parts by comparing the detected locations with locations selected by humans. The parts and detection error rates by the simple hierarchy and by the semantic hierarchy are shown in Table 1. For this test, we additionally used female images from the JAFFE dataset [11]. For this dataset, the classifier based on both simple and semantic hierarchies reached zero class recognition error, but the part detection error rates for the simple hierarchy were significantly higher, demonstrating the difficulty of part detection and the advantage of the semantic hierarchy in part detection.

Part	Simple hierarchy	Semantic hierarchy
	39.1%	25.7%
	41.9%	36.0%
	51.8%	19.2%
	4.87%	4.3%
	18%	3.4%
	28.85%	5.14%
	0.93%	0%
	22.5%	4.2%
	3.2%	0%

Table 1. Percentage of incorrectly detected or missed parts.

In addition to these specific parts, we also compared the average detection rates of parts at different hierarchical levels for the simple and semantic hierarchies. For first-level parts, average False Alarms rate decrease from 16.93% for simple hierarchy to 6.45% for semantic hierarchy, the corresponding hit rate increased from 83.0% to 93.25%. For second-level parts, False Alarms rate decreased from 37.42% for simple hierarchy to 17.74% for semantic hierarchy, the corresponding hit rates increased from 62.6% to 82.2%. The large improvement in the detection of parts and sub-parts arises from the capacity of the scheme to discover during learning the range of different appearances associated with different object parts.

Finally, we compared the performance of the semantic hierarchy on the horses dataset, with the performance of a hierarchy with the same features, but where all the features were treated as conditionally independent given their parent features. The performance of the semantic hierarchy was considerably higher, equal error rate (EER) dropped from 16% to 13%.

5. Discussion

In this paper, a novel representation for object classification and part interpretation, called semantic hierarchy, was presented. It is the first scheme to extract and use semantic parts in feature hierarchies. Unlike previous models, the current scheme automatically detects and represents multiple appearances of the same object part, at all levels in the hierarchy, together with their statistical dependencies. In previous hierarchical schemes, features representing different appearances of the same object part were either not used, or treated as conditionally independent. This independence assumption clearly does not hold in this case, since different appearances of the same part are mutually exclusive, and therefore not independent, even when conditioned on the parent feature. The current scheme avoids this limitation by representing different appearances of the same part by different values of the corresponding variable, thereby ensuring mutual exclusivity. The advantages of the semantic hierarchy over previous hierarchical models include better recognition rate and a large improvement in the detection and localization of ambiguous object parts at lower levels of feature hierarchy.

A novel method was introduced for the difficult task of identifying semantically equivalent object parts. Unlike previous methods, the proposed scheme uses the entire hierarchy as a context to identify semantically equivalent object parts. As a result, the algorithm can find semantic equivalents of features at all levels in the hierarchy simultaneously, using a unified and computationally efficient process.

Object recognition involves more than naming and localizing objects in the scene. As we learn to recognize objects, we also learn to recognize many of their parts and sub-parts, and we can identify the different appearances of parts, such as different hairlines, smiling, neutral or open mouth, different shapes of car headlights etc. The semantic hierarchy described in this paper automatically constructs a representation, which was shown to be useful for

achieving such capabilities as a part of the recognition scheme.

Acknowledgment

This work was supported by ISF Grant 7-0369 and EU IST Grant FP6-2005-015803, and conducted at the Moross Laboratory for Vision and Motor Control.

References

[1] F. Scalzo, J. Piater "Statistical Learning of Visual Feature Hierarchies", IEEE Workshop on Learning in CVPR 2005

[2] G. Bouchard, B. Triggs, “Hierarchical Parts-Based Visual Object Categorization”, CVPR 2005

[3] B. Epshtein, S. Ullman, “Feature Hierarchies for Object Classification”, ICCV, 2005

[4] T. Serre, L. Wolf, T. Poggio, “Object Recognition with Features Inspired by Visual Cortex”, CVPR 2005

[5] Y. LeCun, B. Boser, J.S. Denker, D. Henderson, R.E. Howard, W. Hubbard, L.D. Jackel, “Backpropagation applied to handwritten zip code recognition”, *Neural Comp. 1 (4),* 541-551, 1989.

[6] B. Epshtein, S. Ullman, “Identifying semantically Equivalent Object Fragments”, CVPR, 2005

[7] J. Pearl, “Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference”, Morgan Kaufmann, 1988.

[8] A. Dempster, N. Laird, D. Rubin. "Maximum likelihood from incomplete data via the EM algorithm". *Journal of the Royal Statistical Society*, Series B, 39(1):1–38, 1977

[9] R. Neal, G. Hinton,. "A view of the EM algorithm that justifies incremental, sparse, and other variants". In Michael I. Jordan (editor), *Learning in Graphical Models* pp 355-368. Cambridge, MA: MIT Press, 1999

[10] www.vision.caltech.edu/Image_Datasets/Caltech101/Caltech101.html

[11] http://www.kasrl.org/jaffe.html

[12] A. Gelman, J. Carlin, H. Stern, D. Rubin. “Bayesian Data Analysis”, 2nd edition. CRC Press, 2003

Figure 3. Examples of class images. Rows, from top to bottom: Horses, motorbikes, cars, JAFFE dataset.

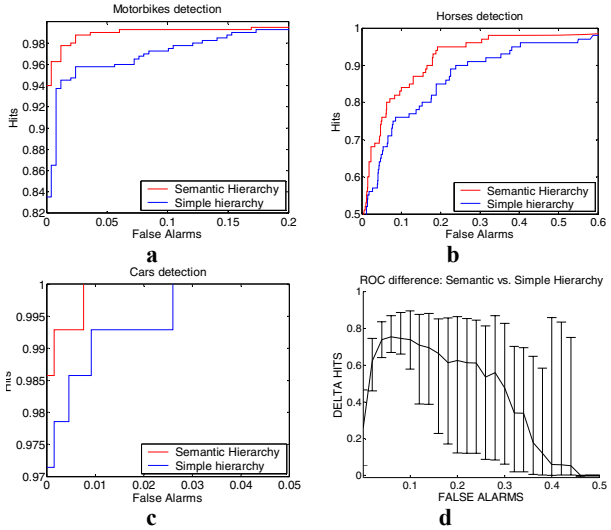

Figure 4. ROC curves for (a) motorbike, (b) horses and (c) cars recognition. Red: Classifier performance based on semantic hierarchy. Blue: Classifier performance based on simple hierarchy. (d): Difference in ROC curve (added hits as a function of false alarms, (mean and sd, 15 runs) between the classifier based on semantic hierarchy and classifier based on simple hierarchy) on motorbikes dataset.

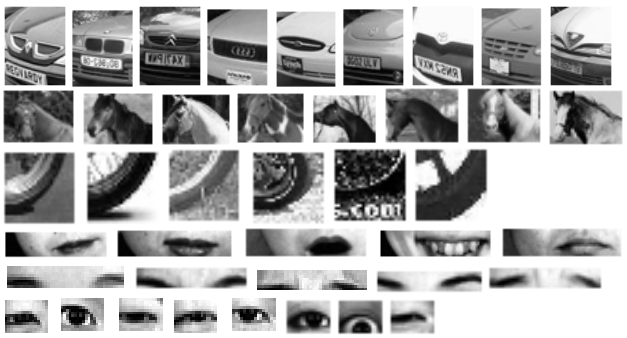

Figure 5. Variability of the object parts detected by the semantic hierarchy.

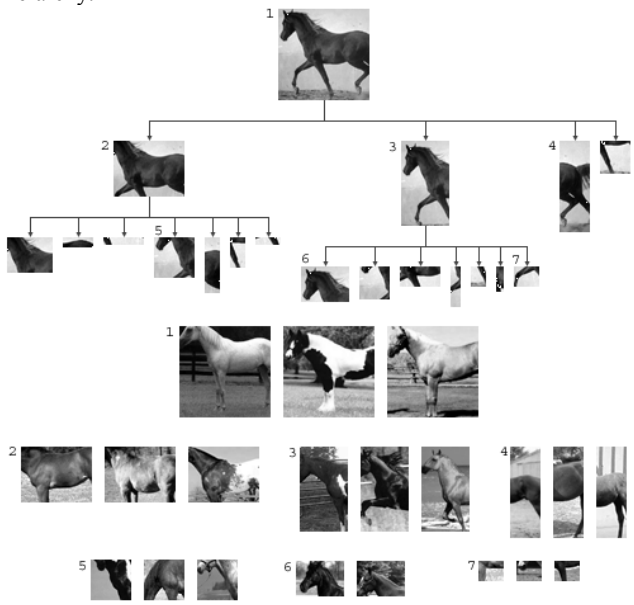

Figure 6. An example of simple hierarchy (top) and examples of additional semantic features at different levels of the semantic hierarchy.

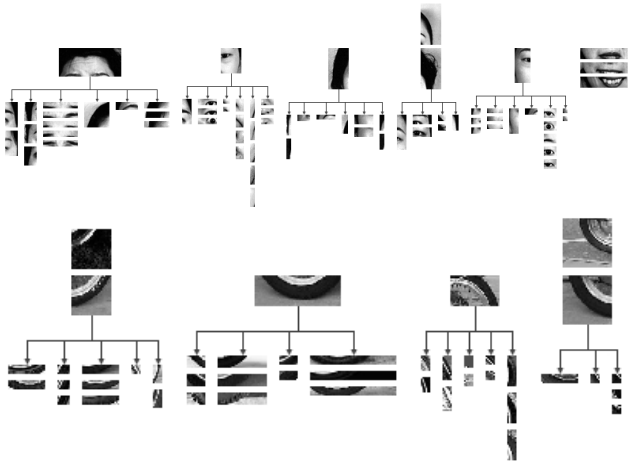

Figure 7. Additional examples of the semantic hierarchies.