Hierarchical Statistical Learning of Generic Parts of Object Structure*

Sanja Fidler, Gregor Berginc, and Aleš Leonardis
Faculty of Computer and Information Science
University of Ljubljana, Slovenia
{sanja.fidler, gregor.berginc, ales.leonardis}@fri.uni-lj.si

Abstract

With the growing interest in object categorization various methods have emerged that perform well in this challenging task, yet are inherently limited to only a moderate number of object classes. In pursuit of a more general categorization system this paper proposes a way to overcome the computational complexity encompassing the enormous number of different object categories by exploiting the statistical properties of the highly structured visual world. Our approach proposes a hierarchical acquisition of generic parts of object structure, varying from simple to more complex ones, which stem from the favorable statistics of natural images. The parts recovered in the individual layers of the hierarchy can be used in a top-down manner resulting in a robust statistical engine that could be efficiently used within many of the current categorization systems. The proposed approach has been applied to large image datasets yielding important statistical insights into the generic parts of object structure.

1 Introduction

Humans classify and recognize a vast number of objects varying in shape, color, size and pose with high accuracy and an apparent ease. It has been a common desire of vision researchers to endow computers with a similar trait. However, the complexity underlying the categorization process has, to a large extent, hindered the success of approaches tackling this difficult task.

The current state-of-the-art categorization methods [11, 16, 4, 5, 25, 20] rely on classification upon a moderate number of object specific features [11, 25, 4] or massive codebooks of local image patches [16], therefore requiring a computationally demanding matching of image features against a large number of stored prototypical ones. While these methods give excellent results for a moderate number of objects, they would have difficulties handling a larger number of object categories that are in the order of tens of thousands [7].

In pursuit of a more general categorization system many authors emphasize the notion of *parts* or shared features to achieve more compact object representations and better generalization properties [9, 8, 15]. However, features shared by a variety of different categories imply an inefficient verification stage with each feature evoking a large set of object hypotheses. An adequate and generally accepted solution to this problem is a hierarchical organization of part combinations increasing in complexity and specificity [23].

The majority of existing hierarchical methods use static parts in the form of feed-forward convolutions, and are consequently only applicable to very specific object classes, i.e., hand-written characters [14] or paper-clip objects [22]. In order to cope with a large variety of object categories, parts should be *learned* from the masses of image data, the idea which dates back to the seminal work of [3] emphasizing the importance of unsupervised acquisition of patterns regularly appearing in images. Moreover, there are strong implications that the human visual system is driven by these principles as well [12].

However, the complexity of unsupervised learning of patterns is known to be exponential and becomes even more intractable at higher levels of feature combinations thus forcing many authors to perform learning on hand-labelled object parts [13, 6] or employ the classical Gestalt principles, which are theorized as being probabilistically most plausible ones [18].

Attempts have been made to perform unsupervised statistical learning of parts. The authors of [24] propose a method for unsupervised hierarchical learning of feature combinations but are mainly concerned with constructing discriminative object specific visual hierarchies designed exclusively for recognition and not categorization purposes. A combinatorial problem is reported even though only a small number of features is used as the starting point. This approach is inherently different from our method, which

*This research has been supported in part by: Research program P2-0214 (RS), EU FP6-004250-IP project CoSy, EU FP6-511051-2 project MOBVIS, SI-A project, and A-SI-CZ project CONEX.

0-7695-2597-0/06 $20.00 © 2006 IEEE

aims at designing one hierarchy of learned *generic object parts*.

In [8, 1] the authors use vector quantization over outputs of SIFT-like descriptors to obtain the parts, while [27] employs K-means clustering of various filter responses collected over the training images. However, the obtained parts are limited to textured images not having the ability to generalize to a large variety of textureless object categories, for which shape is the primary cue, e.g., hand-drawn objects and silhouettes. A step in this direction has been made by [2] by finding edge parts which have been shown effective for face detection. Yet the obtained features are very local and carry too little discriminative information to be used with a larger number of object classes.

The goal of this paper is to propose a novel approach to the problem of parts, specifically to the parts of object structure. We will show how to defy the combinatorial problem of unsupervised pattern acquisition within a hierarchical framework and how the favorable statistics of images can be exploited in order to manage the otherwise prohibitive complexity of a higher-level feature combination. Furthermore, the proposed hierarchy enables the use of a top-down mechanism in each layer resulting in a closed-end robust statistical engine that could be efficiently used within many of the current recognition systems.

The paper is organized as follows: in Section 2 we provide the motivation for the proposed approach which is then presented in Section 3. The implementation details are given in Section 4. The results obtained on various image datasets are shown in Section 5. The paper is concluded with a discussion in Section 6.

2 A Hierarchical Learning Architecture

The importance of structure has often been emphasized in literature as one of the strongest cues for object categorization [9, 4, 5, 25, 20, 15] and is also the focus of our research. Specifically, the proposed approach addresses the issues of constructing a general, structure-based, categorization system that would be capable of recognizing a large number of object classes.

We start with an outline of the properties that such a system should possess:

Hierarchical organization. The previous work done on the analysis of complexity underlying the categorization process implies a hierarchical organization of the system. The arguments can be shortly summarized as follows. The parts used within the categorization system should, on the one hand, enable a computationally feasible matching against image features in the recognition stage implying they should be moderate in number and not too complex. On the other hand, having a small set of parts, shared across categories, would evoke a large number of object hypotheses implying an inefficient verification stage. The most plausible solution to these largely contradictory requirements is a multi-layered system of increasingly complex and specific parts.

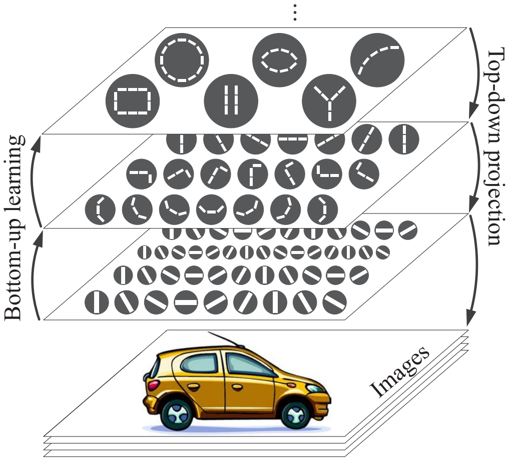

Figure 1. The hierarchical top-down and bottom-up architecture

Top-down projection. In order to achieve robustness against clutter, the parts comprising the individual layers of the hierarchy should be designed in a way to enable a robust verification of the presence of their underlying components.

Bottom-up learning. Parts and their higher level combinations should be learned in an unsupervised manner (at least in the first stages of the hierarchy) in order to avoid hand-labelling of massive image data as well as to capture the regularities within the visual data as effectively and compactly as possible.

As most of the literature might agree with the aforementioned postulates, the enormous combinatorial complexity that comes with the last postulate has slowed the progress on this topic. Our approach aims to overcome this challenging problem.

However, before the learning issues can be addressed, the parts must first be defined with great care. The following are the generally acknowledged properties of parts:

Locality. The receptive fields in which parts are formed should not be too large due to computational complexity of feature combination. Moreover, features that appear close together are likely to belong to the same entity.

Shift and rotation invariance. Since the objects can appear in any position and orientation within an image, the recognition should proceed independently of position and orientation. However, rotation invariance should not be implemented in a way to lose the already limited information of local structure, thus the orientation should rather be separately encoded and accounted for.

Encoded geometrical structure. Parts should encode geometrical relations of their underlying components in order to avoid the so-called binding problem [19] which arises when only the presence of features is reported. However, the relations must be sufficiently loose to account for some degree of variation.

2.1 Complexity Issues

A comprehensive analysis of complexity encompassing recognition of a large number of object classes was done in [26], while the complexity issues of top-down matching were addressed in [10]. We thus only embark on the complexity conveyed by unsupervised learning of higher-level feature combinations.

To form a rough idea, let us consider the following simplified case. Let N be the number of different parts, O the number of orientations the parts can attain and k the number of possible discretized locations with an additional assumption that only one part can appear at each location. The number of all possible combinations in this case is $(1+N\cdot O)^k$ giving a total of 7 billion possibilities when taking values as low as $N=2$, $O=8$ (angle sensitivity being 45 degrees) and $k=8$ locations. While all the combinations might not even appear in images, the computational load of unsupervised learning may still be unmanageable.

Complexity issues aside, a system establishing each combination as a separate part would also have no generalization power which is crucial for successful categorization.

It is therefore apparent that some approximations are needed that would yield a moderate number of parts yet still be capable of representing the majority of data. The idea behind our approach is to find features that appear frequently in images which are therefore optimal for describing the data.

3. Our Approach

The general procedure can be roughly described as follows. The hierarchy is built layer by layer starting with the fixed Layer I comprised of a bank of oriented filters. In the search for the parts of the second layer, parts of Layer I are projected in a top-down manner onto a large data set of images (at this stage this is a simple image convolution). We must emphasize that we are mainly interested in shape information, so we will not be working directly with the values of filter outputs. A certain activation of the filter will simply imply the presence or absence of a local edge segment. To achieve scale invariance, the top-down projection is carried out on every scale by iteratively smoothing and resampling a given image. Bottom-up learning is performed by collecting the statistics of all possible local configurations of parts. The most significant configurations consequently define Layer II. This interplay of top-down and bottom-up mechanism is propagated through the hierarchy and is summarized in Algorithm 1 and depicted in Figure 1 with details of both, top-down and bottom-up, procedures given in subsections 3.2 and 3.3.

Algorithm 1 : A hierarchical learning architecture

```
1: Top-down projection of parts defining Layer I (oriented filters)
2: Bottom-up statistical learning of local configurations of parts of Layer I
   → Result: Parts for Layer II
3: Top-down projection of Layer II parts
4: Bottom-up learning using parts of Layer II
   → Result: Parts for Layer III
       ⋮
```

3.1 Definition of Parts

In accordance with requirements set in Section 2 we define the parts in the following way. Let $\mathcal{P}_i^n$ denote the i-th part in Layer n. Each part $\mathcal{P}_i^n$ is characterized by the center of mass, orientation and a list of subparts (parts of the previous layer) with their respective orientations and positions relative to the center and orientation of $\mathcal{P}_i^n$. Specifically, a $\mathcal{P}_i^n$ that is normalized to the orientation of 0 degrees and has a center in $(0,0)$ encompasses a list $\{(\mathcal{P}_j^{n-1}, \alpha_j, [x_j, y_j]^T, \sigma_j)\}_j$, where α_j and (x_j, y_j) denote the relative orientation and position of $\mathcal{P}_j^{n-1}$, respectively, while σ_j denotes the allowed variance of its position around (x_j, y_j). The translation and rotation of $\mathcal{P}_i^n$ by (x,y) and β, respectively, would therefore imply the following transformation of its subparts: $\{(\mathcal{P}_j^{n-1}, \alpha_j + \beta, [x,y]^T + Rot(\beta)[x_j, y_j]^T, \sigma_j)\}_j$, where $Rot(\beta)$ denotes the rotation matrix by angle β.

For the ease of reference let *what* denote the information about the type of parts, i.e., $\mathcal{P}_j^i$, let *orientation* correspond to the orientation information of parts and let *where* refer to the position information of parts.

3.2 Top-down mechanism

The top-down mechanism consists of two stages, namely the projection and the part selection stage.

3.2.1 Projection Stage

The projection of the n-th Layer parts onto the parts of the preceding layer that were recovered in the processed image proceeds as follows. Each 'activated' part $\mathcal{P}^{n-1}$ within an

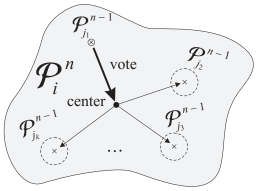
$\mathcal{P}_{j_1}^{n-1}$
vote
$\mathcal{P}_{j_2}^{n-1}$
$\mathcal{P}_i^n$
center
$\mathcal{P}_{j_k}^{n-1}$
$\mathcal{P}_{j_3}^{n-1}$
...

Figure 2. Voting of subparts for part $\mathcal{P}_i^n$

image votes for the presence of all parts $\mathcal{P}^n$ that contain $\mathcal{P}^{n-1}$ in their list of subparts. Specifically, it is the center and orientation of the part that is voted for. Every hypothesized center then checks for the presence of all corresponding subparts of $\mathcal{P}^n$. All fully verified hypotheses are passed to the selection stage. In the case of no complete hypotheses, lower layers of the hierarchy are consulted for verification. The algorithm is described in Alg. 2 and depicted in Fig. 2.

Algorithm 2 : Top-down

```
1: for each subpart $\mathcal{P}^{n-1}$ found in the image do
2:    for each part $\mathcal{P}^n$ containing $\mathcal{P}^{n-1}$ do
3:       Vote for the center and orientation of $\mathcal{P}^n$
4:       According to the hypothesized center check for all
         subparts comprising $\mathcal{P}^n$ (variance in the position
         is allowed)
5:    end for
6:    All fully verified parts $\mathcal{P}^n$ are passed to step 8
7: end for
8: Part selection with the MDL principle
```

3.2.2 Part Selection with the Minimum Description Length Principle

The projection stage results in redundant descriptions of objects. To greatly alleviate the computational cost of bottom-up learning, redundancy of the obtained parts is first reduced by applying a Minimum Description Length (MDL) model-selection method as proposed in [17], but tailored to our specific models.

We shortly summarize the procedure. By considering each part as a model with a certain cost and error, the objective function that selects an optimal subset of parts which fully describe the object yet discards the redundant parts can be written in the form

$$F(\mathbf{m}) = \mathbf{m}^T Q \mathbf{m} \tag{1}$$

where $\mathbf{m} = [m_1, \ldots, m_N]$ and m_i denotes the *presence variable* taking value 1 for the presence and 0 for the absence of part $\mathcal{P}_i^n$ in the final description. The diagonal terms of the matrix Q stand for the cost-benefit value of a corresponding part $\mathcal{P}_i^n$ while the off-diagonal terms express the relations of the overlapping parts. The objective function is solved using the greedy algorithm as proposed in [17]. The details are given in Section 4.

This procedure reduces the original number of activated parts by a factor of 5 to 10 by losing no information with respect to the description of the object.

3.3 Bottom-up learning

We first present the conceptual ideas behind the proposed learning process and give the implementation details in the next section.

Suppose that parts of Layer $n-1$ have already been obtained and the bottom-up construction of parts of Layer n is under consideration. To meet the requirements set in Section 2 local part configurations will be investigated by collecting statistics over a large dataset of images. Furthermore, in order to achieve shift and rotation invariance the part-centered coordinate system is chosen (depicted in utmost left of Fig. 3). This means that for each part $\mathcal{P}_i^{n-1}$ activated in any processed image, its local neighborhood of the surrounding parts is first normalized with respect to orientation and position of the central part $\mathcal{P}_i^{n-1}$, and stored for further processing.

In line with the conclusion of 2.1 only the most frequent local arrangements are sought. These will consequently define parts in Layer n. However, keeping track of all possible local configurations (or at least a vast majority of them) seems, according to complexity issues addressed in 2.1, an elusive task. Next, we show how to overcome this problem by decoupling the complexity into the sequential statistical analysis of different aspects of feature combinations.

Local combinations encode all, $what + orient. + where$ information (see 3.1 for notation). Therefore, if a particular combination of only one type of information (for example *what*) appears with small probability in images, then adding further information can only reduce the total probability of the corresponding configurations. Since only the most frequent configurations are sought, such combinations can be discarded from further analysis.

Thereby the bottom-up learning as we propose it proceeds as follows. An image dataset is processed and the statistics of all *what* configurations are collected. Only the most frequent configurations are passed on to the next stage, where the images are processed again and all possible *orientation* combinations in which the previously selected *what* features appear are stored. The most frequent ones are selected resulting in configurations now encoding *what* and *orientation* information. Finally, the statistics of all the relative locations where these features occur in are collected and the significant peaks in certain locations are then added to arrive at the final parts. Alg.3 and Fig.3 summarize the procedure.

Algorithm 3 : Bottom-up learning

```
for each image scale do
    for each part $\mathcal{P}^{n-1}$ activated in an image do
        Local neighborhood is defined and normalized with respect to $\mathcal{P}^{n-1}$
        Complexity is decoupled and separate statistics are sought:
        what parts (only most frequent cases are passed on to the following stage)
        orientation parts (only most frequent cases are passed on to the following stage)
        where parts
    end for
end for
```

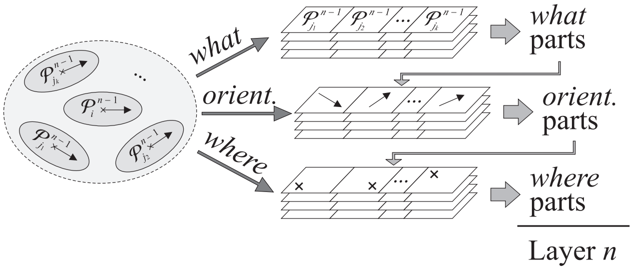
what
orient.
where
$\mathcal{P}_{j_k}^{n-1}$
$\mathcal{P}_i^{n-1}$
$\mathcal{P}_{j_1}^{n-1}$
$\mathcal{P}_{j_2}^{n-1}$
$\mathcal{P}_{j_1}^{n-1}$ $\mathcal{P}_{j_2}^{n-1}$ ... $\mathcal{P}_{j_k}^{n-1}$
what parts
orient. parts
where parts
Layer n

Figure 3. Bottom-up learning procedure of local configurations (left) by decoupling the complexity in separate statistics.

The following subsections give the details of how each statistic is carried out.

3.3.1 Determining the Neighborhood Size

Since the number of possible combinations grows exponentially with the size of the local neighborhood under consideration, the size must be chosen with great care. Several different sizes must be tested by processing the images and collecting *count* histograms H_j (an element $H_j(i)$ is incremented whenever i parts are found in the local neighborhood of $\mathcal{P}_j^{n-1}$).

The final size of the neighborhood consequently used in further analysis is set to the size that produces the most manageable histograms, i.e., histograms that have peeks in lower indices and contain considerably lower values thereon.

Denote the index of the last significant peak in the corresponding histograms by m. This implies that keeping track of a limited number of the activated parts in the chosen size of the local neighborhood will not result in losing too much information.

3.3.2 Statistics of *What*

Let N be the number of all parts defining Layer $n-1$. For each type of part, $\mathcal{P}^{n-1}$, activated in an image a separate *what* statistics of its local neighborhood is collected in the form of vectors (and stored in a matrix):

$$[\underbrace{a_{11}\ a_{12}\ \ldots\ a_{1m}}_{\mathcal{P}_1^{n-1}}\ |\ \ldots\ |\ \underbrace{a_{N1}\ a_{N2}\ \ldots\ a_{Nm}}_{\mathcal{P}_N^{n-1}}\ |\ \underbrace{a}_{\text{no part}}], \tag{2}$$

where $a_{ij} = 1$ if the receptive field contains j parts of the type $\mathcal{P}_i^{n-1}$, with $a_{ij} = 0$ otherwise. If the local neighborhood contains no other subparts except the central one, all a_{ij} are set to 0 and a is set to 1.

The significant configurations are selected in the following way. By adding up the elements across each column in the matrix, the columns with significant sums are selected. For each selected column with the corresponding index i, a submatrix that has ones in the i-th column is processed in a similar manner, yielding another set of significant columns. This procedure is repeated until all the sums in the remaining columns drop to the value by a certain percentage lower than the sum of the starting column i. The obtained sequences of columns represent the final *what* parts. We must emphasize that the configurations chosen are conditional on the central part, i.e., they take the form $(\mathcal{P}_i^{n-1}, \{\mathcal{P}_{i_1}^{n-1}, \ldots, \mathcal{P}_{i_K}^{n-1}\})$.

3.3.3 Statistics of *Orientation*

Following the idea of the previous subsection the orientation information can be added to each of the selected *what* parts by a similar procedure. Whenever a certain neighborhood within an image contains any of the previously selected *what* parts their corresponding orientations are stored in vectors:

$$[\underbrace{b_{11}\ b_{12}\ \ldots\ b_{1O}}_{\mathcal{P}_{i_1}^{n-1}}\ |\ \ldots\ |\ \underbrace{b_{K1}\ b_{K2}\ \ldots\ b_{KO}}_{\mathcal{P}_{i_K}^{n-1}}], \tag{3}$$

where O is the number of orientations in which each part $\mathcal{P}_{i_k}^{n-1} \in what$ can appear in, with $b_{ko} = 1$ if $\mathcal{P}_{i_k}^{n-1}$ occurs in the orientation of $2\pi(o-1)/O$ relative to the orientation of the central part $\mathcal{P}_i^{n-1}$. Since certain type of parts within the *what* part under consideration can be the same, the ambiguity is resolved by the lexicographical ordering of parts with respect to their positions in the local neighborhood.

The most significant orientations are found by employing the procedure described in the previous subsection. The final features now encode both, *what* and *orient.* information, and take the form $(\mathcal{P}_i^{n-1}, \{(\mathcal{P}_{i_1}^{n-1}, \alpha_{i_1}), \ldots, (\mathcal{P}_{i_K}^{n-1}, \alpha_{i_K})\})$.

3.3.4 Statistics of *Where*

The statistical analysis of *what* is carried out in a similar fashion as the previous ones, requiring another pass through the images.

Whenever a certain previously selected $what + orientation$ configuration is encountered in a local neighborhood, the positions of parts relative to the central one are stored as rows in a matrix:

$$\left[\begin{array}{c|c|c} \underset{(\mathcal{P}_{i_1}^{n-1},\alpha_{i_1})}{x_1, y_1} & \dots & \underset{(\mathcal{P}_{i_K}^{n-1},\alpha_{i_K})}{x_K, y_K} \end{array}\right], \tag{4}$$

For each feature, clustering of positions is performed using the corresponding columns and clusters with significant peeks are selected for further processing. For each cluster all rows of the matrix that have (x, y) within the cluster and belong to the corresponding feature are selected. The obtained submatrix is again subjected to clustering. This process is continued until no significant clusters are found.

The final features now take the form $(\mathcal{P}_i^{n-1}, \{(\mathcal{P}_j^{n-1}, \alpha_j, (x_j, y_j), \sigma_j)\})$ and thus encode all aspects of feature information. As such they are pronounced as parts of Layer n. Note that the variances σ_j are also learned (denoting the radius of clusters). Furthermore, for each part the center of mass and orientation is calculated and its subparts are normalized accordingly.

4 Implementation details

4.1 Layer I

The Parts. The first layer in the hierarchy consists of a family of Gabor filters:

$$g_{\lambda,\psi,\varphi}(x, y) = e^{-\frac{u^2+\gamma^2 v^2}{2\sigma^2}} \cos\left(\frac{2\pi u}{\lambda} + \varphi\right)$$
$$u = x\cos\psi - y\sin\psi, \quad v = x\sin\psi + y\cos\psi,$$

where (x, y) represents the center of the filter's receptive field, with the parameters set as in [21]. We use the set of two filter banks, one with even ($\varphi = 0$) and the other with odd ($\varphi = -1/2\pi$) Gabor kernels with both banks containing filters in 6 equidistant orientations ($\psi = i(\pi/6)$, $i = 0, 1, \dots, 5$).

Top-down: projection. After an image is convolved with the filter banks, the total energy for each orientation is computed [21]:

$$E_{\lambda,\psi}(x, y) = \sqrt{r^2_{\lambda,\psi,0}(x, y) + r^2_{\lambda,\psi,-\pi/2}(x, y)}, \tag{5}$$

where $r_{\lambda,\psi,0}(x, y)$ and $r_{\lambda,\psi,-\pi/2}(x, y)$ are responses of even and odd Gabor filters, respectively. In order to find the best orientation of a local edge centered in point (x, y) a maximum of all orientations in $E_{\lambda,\psi}(x, y)$ is computed, similarly as in [22]. Since we are only interested in presence or absence of a local edge, only points above a certain threshold are kept. Furthermore, local maxima are found by selecting only those points having at most two surrounding points in a 3 × 3 window with larger response values. This is the heuristic we have chosen to use and it has proven successful. Every selected point represents the center of an local edge.

Figure 4. Typical examples of images.

Top-down: selection. In the first Layer we only deal with one type of part, a line segment, and the MDL selection of [17] is adapted to account for models in the form of local oriented lines.

4.2 Layer II and III

The only procedure that needs further explanation is **Top-down: selection.** The cost of a part $\mathcal{P}^n$ is defined as $-\log P(\mathcal{P}^n)$ where P refers to the prior probability with which $\mathcal{P}^n$ appears as a local configuration within an image. This probability is obtained within the proposed statistical analysis of images. This choice of cost implies that those parts having a high probability of occurrence and therefore having shorter codes offer more efficient representation of the data and should thus be selected more often. The costs of intersecting parts are treated similarly as in [17].

5 Results

The proposed hierarchical learning framework was applied to three different image datasets, namely the clipart dataset[1] containing 2394 images(Fig. 5(a) and (b)) and two Caltech object categories (1027 airplane and 826 motorbike images - Fig. 5(c) and (d), respectively)[2]. To show the quick convergence of the statistics we additionally inspected a small subset of images in the clipart dataset.

The size of the local neighborhood to be used for learning the parts of Layer 2 was investigated first with the resulting optimal size being twice the size of filter's receptive field. Since the first Layer of the hierarchy contains parts of only one type (a line segment) the count statistics as described in 3.3.1 corresponds to the *what* statistics of 3.3.2 and is depicted in Fig. 5(a). The statistics were collected for each image scale separately showing that they are almost the same for all scales, leading us to the conclusion that we can merge statistics of different scales (see the last

[1] http://www.barrysclipart.com/barrysclipart.com/index.php

[2] http://www.robots.ox.ac.uk/ vgg/data3.html

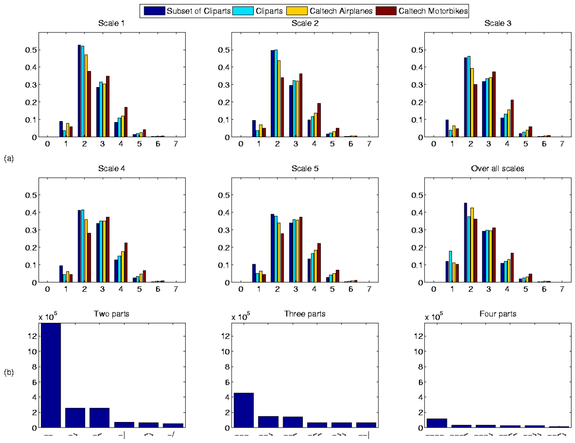

Figure 5. (a) The What histograms for different image datasets and scales. (b) The Orientation histograms for two, three and four parts on scale 1 of the Clipart dataset. The slopes in labels indicate the orientations of parts.

bar graph in Fig. 5(a)). Furthermore, it can be seen that only neighborhood containing one to at most four parts have significant peaks and need to be explored further.

Next, Fig.5(b) shows the obtained *orientation* histograms corresponding to the neighborhoods containing two, three and four parts (only the first few of the selected, most frequent, orientations are shown). The fast drop in frequency is even more apparent than for the *what* features.

Table 5 shows the number of all *orientation* combinations that were selected and passed on to the next, the *where* stage.

Image dataset	# of selected combinations
Subset of Cliparts	150
Cliparts	141
Airplanes	67
Motorbikes	379

Table 1. Number of selected what+orient. combinations for different datasets

The final parts for Layer II are presented in Fig. 7 with the learned variances of corresponding subparts depicted in the second row. The third row contains probabilities of occurrence with respect to all local configurations that appeared within the image datasets. The overall probability that all the selected parts represent is shown in Fig. 5. It can be seen that the number of all possible local configurations reduces to as little as 5 parts representing the vast majority (over 96%) of occurred configurations.

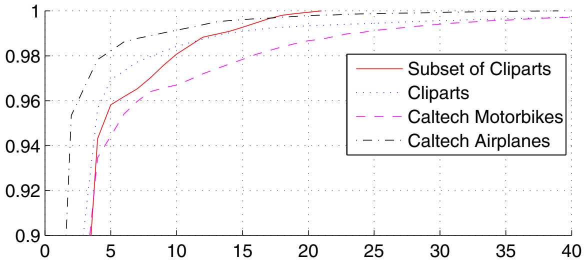

Figure 6. The percentage of occurrence of Layer 2 features for different datasets.

Finally, Fig. 8 depicts a few examples of the final parts for Layer III. These results, together with the results in Fig. 7 can also be seen as a statistical confirmation to some of the well known Gestalt principles.

6 Summary and conclusions

In this paper we presented a novel approach that enables unsupervised learning of generic parts of object structure within a hierarchical framework by exploiting the regularities present in the visual data. The approach additionally enables the use of a top-down mechanism resulting in a closed-end robust statistical engine that could be efficiently used within many of the current categorization systems.

The results obtained on a large dataset of images confirm the presumed convergent statistics of natural images and additionally confirm a number of Gestalt principles that have so-far been only theorized as statistically most plausible ones.

Our future work will include designing further layers that will enable object categorization, with the ability to deal with a significantly larger number of object categories than the current categorization methods.

References

[1] A. Agarwal and B. Triggs. Hyperfeatures - multilevel local coding for visual recognition. Research Report RR-5655, INRIA, 2005.

[2] Y. Amit and D. Geman. A computational model for visual selection. *Neural Computation*, 11(7):1691–1715, 1999.

[3] H. B. Barlow. Conditions for versatile learning, helmholtzs unconscious inference, and the task of perception. *Vision Research*, 30:1561–1571, 1990.

[4] S. Belongie, J. Malik, and J. Puzicha. Shape matching and object recognition using shape contexts. *IEEE PAMI*, 24(4):509–522, 2002.

[5] A. C. Berg, T. L. Berg, and J. Malik. Shape matching and object recognition using low distortion correspondences. In *CVPR (1)*, pages 26–33, 2005.

0-7695-2597-0/06 $20.00 © 2006 IEEE

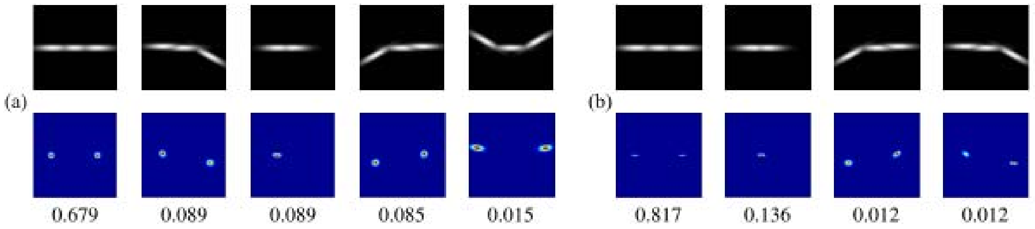

Figure 7. The first row depicts the final parts comprising Layer II obtained for (a) Cliparts and (b) Airplanes. The variances of position distributions of parts, relative to the central part, are depicted in the middle. The feature probabilities are listed in the last row.

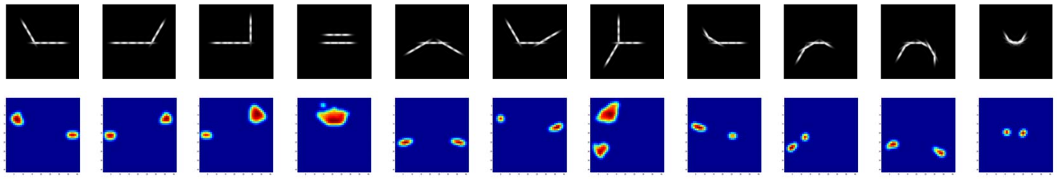

Figure 8. (a) Examples of Layer 3 parts, (b) variances of positions of the surrounding subparts

[6] M. Burl, M. Weber, and P. Perona. A probabilistic approach to object recognition using local photometry and global geometry. In *ECCV-B 98*, pages 628–641, 1998.

[7] J. C. Corbeil and A. Archambault. *The Firefly Visual Dictionary*. Toronto : Firefly Books, 2002.

[8] W. F. E. Sudderth, A. Torralba and A. Willsky. Learning hierarchical models of scenes, objects, and parts. In *ICCV*, pages 1331–1338, 2005.

[9] S. Edelman, N. Intrator, and J. S. Jacobson. Unsupervised learning of visual structure. In *Biologically Motivated Computer Vision*, pages 629–642, 2002.

[10] G. J. Ettinger. Hierarchical object recognition using libraries of parameterized model sub-parts. *Technical report AITR-963*, 1987.

[11] R. Fergus, P. Perona, and A. Zisserman. Object class recognition by unsupervised scale-invariant learning. In *CVPR(2)*, pages 264–271, 2003.

[12] J. Fiser and R. N. Aslin. Statistical learning of new visual feature combinations by infants. *Proc Natl Acad Sci U S A*, 99(24):15822–15826, 2002.

[13] D. Huttenlocher and P. Felzenszwalb. Pictorial structures for object recognition. *IJCV*, 61(1):55–79, 2005.

[14] S. M. K. Fukushima and T. Ito. Neocognitron: a neural network model for a mechanism of visual pattern recognition. *IEEE SMC*, 13(3):826–834, 1983.

[15] S. Krempp, D. Geman, and Y. Amit. Sequential learning of reusable parts for object detection. *Technical report, CS Johns Hopkins*, 2002.

[16] B. Leibe, A. Leonardis, and B. Schiele. Combined object categorization and segmentation with an implicit shape model. In *ECCV'04, SLCV Workshop*, 2004.

[17] A. Leonardis, A. Gupta, and R. Bajcsy. Segmentation of range images as the search for geometric parametric models. *IEEE IJCV*, 14(3):253–277, 1995.

[18] D. G. Lowe. Three-dimensional object recognition from single two-dimensional images. *Artificial Intelligence*, 31(3):355–395, 1987.

[19] B. W. Mel and J. Fiser. Minimizing binding errors using learned conjunctive features. *Neural Computation*, 12(4):731–762, 2000.

[20] R. C. Nelson and A. Selinger. Large-scale tests of a keyed, appearance-based 3-d object recognition system. *Vision Research*, 38(15-16):2469–2488, 1998.

[21] N. Petkov. Biologically motivated computationally intensive approaches to image pattern recognition. *Future Generation Computer Systems*, 11(4-5):451–465, 1995.

[22] M. Riesenhuber and T. Poggio. Hierarchical models of object recognition in cortex. *Nature Neuroscience*, 2(11):1019–1025, Nov. 1999.

[23] D. P. S. Geman and Z. Chi. Composition systems. *Quarterly of Applied Mathematics*, Vol. 60(Nb. 4):707–736, 2002.

[24] F. Scalzo and J. H. Piater. Statistical learning of visual feature hierarchies. In *IEEE Workshop on Learning, CVPR*, 2005.

[25] J. Shotton, A. Blake, and R. Cipolla. Contour-based learning for object detection. In *ICCV*, pages 503–510, 2005.

[26] J. K. Tsotsos. Analyzing vision at the complexity level. *Behavioral and Brain Sciences*, 13(3):423–469, 1990.

[27] J. Winn, A. Criminisi, and T. Minka. Object categorization by learned universal visual dictionary. In *ICCV*, pages 1800–1807, 2005.