A Graph Theoretic Approach for Object Shape Representation in Compositional Hierarchies Using a Hybrid Generative-Descriptive Model

Umit Rusen Aktas*, Mete Ozay*, Aleš Leonardis, and Jeremy L. Wyatt

School of Computer Science, The University of Birmingham, Edgbaston, Birmingham, B15 2TT, UK
{u.aktas,m.ozay,a.Leonardis,j.l.wyatt}@cs.bham.ac.uk

Abstract. A graph theoretic approach is proposed for object shape representation in a hierarchical compositional architecture called Compositional Hierarchy of Parts (CHOP). In the proposed approach, vocabulary learning is performed using a hybrid generative-descriptive model. First, statistical relationships between parts are learned using a *Minimum Conditional Entropy Clustering* algorithm. Then, selection of *descriptive* parts is defined as a frequent subgraph discovery problem, and solved using a Minimum Description Length (MDL) principle. Finally, part compositions are constructed using learned statistical relationships between parts and their description lengths. Shape representation and computational complexity properties of the proposed approach and algorithms are examined using six benchmark two-dimensional shape image datasets. Experiments show that CHOP can employ part shareability and indexing mechanisms for fast inference of part compositions using learned shape vocabularies. Additionally, CHOP provides better shape retrieval performance than the state-of-the-art shape retrieval methods.

1 Introduction

Hierarchical compositional architectures have been studied in the literature as representations for object detection [7], categorization [10,19,21] and parsing [25]. A detailed review of the recent works is given in [26].

In this paper, we propose a graph theoretic approach for object shape representation in a hierarchical compositional architecture, called Compositional Hierarchy of Parts (CHOP), using a hybrid generative-descriptive model. Unlike hierarchical compositional architectures studied in the literature, CHOP enables us to measure and employ generative and descriptive properties of parts for the inference of part compositions in a graph theoretic framework considering part shareability, indexing and matching mechanisms. We learn a compositional vocabulary of shape parts considering not just their statistical relationships but also their *shape description* properties to generate object shapes. In addition, we take advantage of integrated models for utilization of part shareability in

* The first and second author contributed equally.

D. Fleet et al. (Eds.): ECCV 2014, Part III, LNCS 8691, pp. 566–581, 2014.

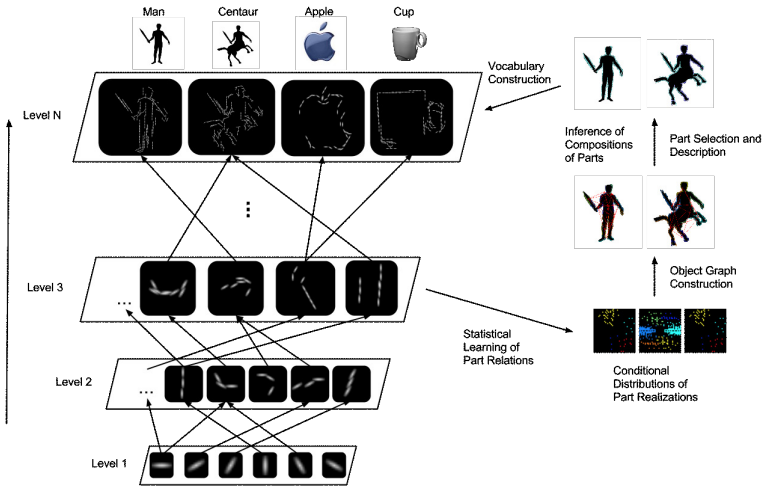


Fig. 1. The information flow of Compositional Hierarchy of Parts (CHOP)

order to construct *dense* representations of shapes in learned vocabularies for fast indexing and matching.

A diagram expressing the information flow in CHOP is given in Fig. 1. At the first layer $l = 1$ of CHOP, we extract Gabor features from a given set of images. We define parts as random graphs and represent part realizations as the instances of random graphs observed on in some dataset. At each consecutive layer, $l \geq 1$, we first learn the statistical relationships between parts using a *Minimum Conditional Entropy Clustering* (MCEC) algorithm [16] measuring conditional distributions of part realizations. For this purpose, we compute the statistical relationship between two parts by measuring the amount of information needed to describe a part realization R_i of a part P_i given the part realization R_j of another part P_j, for all parts represented in a learned vocabulary, and for all realizations observed on images. Using the learned statistical relationships, we represent compositions of object parts as object graphs. Second we define the *contribution* of a part P_i to the representation of a shape by measuring the *conditional description length* of the compositional representation of the shape given the part P_i, using the Minimum Description Length (MDL) principle. In order to select the parts which represent compositional shapes with minimum description lengths, we solve a frequent subgraph discovery problem. Then, part compositions are inferred considering learned statistical relationships between parts and their description lengths. Finally, the inferred part compositions are used to construct shape vocabularies. The steps are recursively employed until no more compositions are inferred.

The paper is organised as follows. Related work and the contributions of the paper is summarized in the next section. The proposed Compositional Hierarchy of Parts (CHOP) algorithm is given in Section 3. Experimental analyses are given in Section 4, and Section 5 concludes the paper.

2 Related Work and Contribution

In [8] and [15], shape models are learned using hierarchical shape matching algorithms. Kokkinos and Yuille [13] first decompose object categories into parts and shape contours using a top-down approach. Then, they employ a Multiple Instance Learning algorithm to discriminatively learn the shape models using a bottom-up approach. However, part-shareability and indexing mechanisms [11] are not employed and considered as future work in [13]. Fidler, Boben and Leonardis [11] analyzed crucial properties of hierarchical compositional approaches that should be invoked by the proposed architectures. Following their analyses, we develop an unsupervised generative-descriptive model for learning a vocabulary of parts considering part-shareability, and performing *efficient* inference of object shapes on test images using an indexing and matching method.

Fidler and Leonardis proposed a hierarchical architecture, called *Learned Hierarchy of Parts* (LHOP), for compositional representation of parts [10]. The main difference between LHOP and the proposed CHOP is that CHOP employs a hybrid generative-descriptive model for learning shape vocabularies using information theoretic methods in a graph theoretic framework. Specifically, CHOP first learns statistical relationships between varying number of parts, i.e. compositions of K-parts instead of the two-part compositions called (duplets) used in LHOP [10,11]. Second, shape descriptive properties of parts are integrated with their statistical properties for inference of part compositions. In addition, the number of layers in the hierarchy are not pre-defined but determined in CHOP according to the statistical properties of the data.

MDL models have been employed for statistical shape analysis [5,24], specifically to achieve compactness, specificity and generalization ability properties of shape models [5] and segmentation algorithms [6]. We employ MDL for the discovery of compositions of shape parts considering the statistical relationships between the parts, recursively in a hierarchical architecture. Hybrid generative-descriptive models have been used in [12] by employing Markov Random Fields and component analysis algorithms to construct descriptive and generative models, respectively. Although their proposed approach is hierarchical, they do not learn compositional vocabularies of parts for shape representation.

Although our primary motivation is constructing a hierarchical compositional model for shape representation, we also examined the proposed algorithms for shape retrieval in the Experiments section. For this purpose, we compare the similarity between shapes using discriminative information about shape structures extracted from a learned vocabulary of parts and their realizations. Theoretical and experimental results of [20,22,23] on spectral properties of isomorphic graphs show that the eigenvalues of the adjacency matrices of two isomorphic graphs are ordered in an interval, and therefore provide useful information for discrimination of graphs. Assuming that shapes of the objects belonging to a category are represented (*approximately*) by isomorphic graphs, we can obtain discriminative information about the shape structures by analyzing spectral properties of the part realizations detected on the shapes.

Our contributions in this work are threefold:

1. We introduce a graph theoretic approach to represent objects and parts in compositional hierarchies. Unlike other hierarchical methods [7,13,25], CHOP learns shape vocabularies using a hybrid generative-descriptive model within a graph-based hierarchical compositional framework. The proposed approach uses graph theoretic tools to analyze, measure and employ geometric and statistical properties of parts to infer part compositions.
2. Two information theoretic methods are employed in the proposed CHOP algorithm to learn the statistical properties of parts, and construct compositions of parts. First we learn the relationship between parts using MCEC [16]. Then, we select and infer compositions of parts according to their shape description properties defined by an MDL model.
3. CHOP employs a hybrid generative-descriptive model for hierarchical compositional representation of shapes. The proposed model differs from frequency-based approaches in that the part selection process is driven by the MDL principle, which effectively selects parts that are both frequently observed and provide *descriptive* information for the representation of shapes.

3 Compositional Hierarchy of Parts

In this section, we give the descriptions of the algorithms employed in CHOP in its training and testing phases. In the next section, we first describe the preprocessing algorithms that are used in both training and testing. Next, we introduce the vocabulary learning algorithms in Section 3.2. Then, we describe the inference algorithms performed on the test images in Section 3.3.

3.1 Preprocessing

Given a set of images $S = \{s_n, y_n\}_{n=1}^{N}$, where $y_n \in \mathbb{Z}^+$ is the category label of an image s_n, we first extract a set of Gabor features $F_n = \{f_{nm}(\mathbf{x}_{nm}) \in \mathbb{R}\}_{m=1}^{M}$ from each image s_n using Gabor filters employed at location $\mathbf{x}_{nm}$ in s_n at Θ orientations [10]. Then, we construct a set of Gabor features $F = \bigcup_{n=1}^{N} F_n$. In this work, we compute the Gabor features at $\Theta = 6$ different orientations. In order to remove the redundancy of Gabor features, we perform non-maxima suppression. In this step, a Gabor feature with the Gabor response value $f_{nm}(\mathbf{x}_{nm})$ is removed from F_n if $f_{nm}(\mathbf{x}_{nm}) < f_{na}(\mathbf{x}_{na})$, for all Gabor features extracted at $\mathbf{x}_{na} \in \aleph(\mathbf{x}_{nm})$, where $\aleph(\mathbf{x}_{nm})$ is a set of image positions of the Gabor features that reside in the neighborhood of $\mathbf{x}_{nm}$ defined by Euclidean distance in $\mathbb{R}^2$. Finally, we obtain a set of suppressed Gabor features $\hat{F}_n \subset F_n$ and $\hat{F} = \bigcup_{n=1}^{N} \hat{F}_n$.

3.2 Learning a Vocabulary of Parts

Given a set of training images S^{tr}, we first learn the statistical properties of parts using their realizations on images at a layer l. Then, we infer the compositions of parts at layer $l+1$ by minimizing the description length of the object

descriptions defined as *Object Graphs*. In order to remove the redundancy of the compositions, we employ a *local inhibition* process that was suggested in [10]. Statistical learning of part structures, inference of compositions and local inhibition processes are performed by constructing compositions of parts at each layer, recursively, and the details are given in the following subsections.

Definition 1 (Parts and Part Realizations).
The i^{th} part constructed at the l^{th} layer $\mathcal{P}_i^l = (\mathcal{G}_i^l, \mathcal{Y}_i^l)$ is a tuple consisting of a directed random graph $\mathcal{G}_i^l = (\mathcal{V}_i^l, \mathcal{E}_i^l)$, where $\mathcal{V}_i^l$ is a set of nodes and $\mathcal{E}_i^l$ is a set of edges, and $\mathcal{Y}_i^l \in \mathbb{Z}^+$ is a random variable which represents the identity number or label of the part. The realization $R_i^l(s_n) = (G_i^l(s_n), Y_i^l(s_n))$ of $\mathcal{P}_i^l$ is defined by 1) *$Y_i^l(s_n)$ which is the realization of $\mathcal{Y}_i^l$ representing the label of the part realization on an image (s_n), and* 2) *the directed graph $G_i^l(s_n) = \{V_i^l(s_n), E_i^l(s_n)\}$ which is an instance of the random graph $\mathcal{G}_i^l$ computed on a training image $(s_n) \in S^{tr}$, where $V_i^l(s_n)$ is a set of nodes and $E_i^l(s_n)$ is a set of edges of $G_i^l(s_n)$, $\forall n = 1, 2, \ldots, N_{tr}$.*

At the first layer $l = 1$, each node of $\mathcal{V}_i^1$ is a part label $\mathcal{Y}_i^1 \in \mathcal{V}_i^1$ taking values from the set $\{1, 2, \ldots, \Theta\}$, and $\mathcal{E}_i^1 = \varnothing$. Similarly, $E_i^1(s_n) = \varnothing$, and each node of $V_i^1(s_n)$ is defined as a Gabor feature $f_{na}^i(\mathbf{x}_{na}) \in \hat{F}_n^{tr}$ observed in the image $s_n \in S^{tr}$ at the image location $\mathbf{x}_{na}$, i.e. the a^{th} realization of $\mathcal{P}_i^l$ observed in $s_n \in S^{tr}$ at $\mathbf{x}_{na}$, $\forall n = 1, 2, \ldots, N_{tr}$. In the consecutive layers, the parts and part realizations are defined recursively by employing layer-wise mappings $\Psi_{l,l+1}$ defined as

$$\Psi_{l,l+1} : (\mathcal{P}^l, R^l, \mathbb{G}_l) \rightarrow (\mathcal{P}^{l+1}, R^{l+1}), \forall l = 1, 2, \ldots, L, \tag{1}$$

where $\mathcal{P}^l = \{\mathcal{P}_i^l\}_{i=1}^{A_l}$, $R^l = \{R_i^l(s_n) : s_n \in S^{tr}\}_{i=1}^{B_l}$, $\mathcal{P}^{l+1} = \{\mathcal{P}_j^{l+1}\}_{j=1}^{A_{l+1}}$, $R^{l+1} = \{R_j^{l+1}(s_n) : s_n \in S^{tr}\}_{j=1}^{B_{l+1}}$ and $\mathbb{G}_l$ is an object graph which is defined next. □

In the rest of this section, we will use $R_j^l(s_n) \triangleq R_j^l$, $\forall j = 1, 2, \ldots, B_l$, $\forall l = 1, 2, \ldots, L$, $\forall s_n \in S^{tr}$, for the sake of simplicity in the notation.

Definition 2 (Receptive and Object Graph).
A receptive graph of a part realization $\mathcal{R}_i^l$ is a star-shaped graph $RG_i^l = (V_i^l, E_i^l)$, which is induced from a receptive field centered at the root node $\mathcal{R}_i^l$. A directed edge $e_{ab} \in E_i^l$ is defined as

$$e_{ab} = \begin{cases} (a^l, b^l, \phi_{ab}^l), & \text{if} \quad \mathbf{x}_{nb} \in \aleph(\mathbf{x}_{na}), a = i \\ \varnothing, & otherwise \end{cases}, \tag{2}$$

where $\aleph(\mathbf{x}_{na})$ is the set of part realizations that reside in a neighborhood of a part realization R_a^l in an image s_n, $\forall R_a^l, R_b^l \in V_i^l, b \neq i$ and $\forall s_n \in S^{tr}$. ϕ_{ab}^l defines the statistical relationship between R_a^l and R_b^l, as explained in the next subsection.

The structure of part realizations observed at the l^{th} layer on the training set S^{tr} is described using a directed graph $\mathbb{G}_l = (\mathbb{V}_l, \mathbb{E}_l)$, called an object graph, where $\mathbb{V}_l = \bigcup_i V_i^l$ is a set of nodes, and $\mathbb{E}_l = \bigcup_i E_i^l$ is a set of edges, where V_i and E_i is the set of nodes and edges of a receptive graph RG_i, $\forall i$, respectively. □

Learning of Statistical Relationships between Parts and Part Realizations. We compute the *conditional* distributions $P_{\mathcal{P}_i^l}(R_a^l|\mathcal{P}_j^l = R_b^l)$ for each $i = Y_a^l$ and $j = Y_b^l$ between all possible pairs of parts $(\mathcal{P}_i^l, \mathcal{P}_j^l)$ using S^{tr} at the l^{th} layer. However, we select a set of modes $\mathcal{M}^l = \{M_{ij} : i = 1, 2, \ldots, B_l, j = 1, 2, \ldots, B_l\}$, where $M_{ij} = \{M_{ijk}\}_{k=1}^K$ of these distributions instead of detecting a single mode. For this purpose, we define the mode computation problem as a *Minimum Conditional Entropy Clustering* problem [16] as

$$Z_{ijk} := \arg\min_{\pi_k \in C} H(\pi_k, R_a^l|R_b^l), \tag{3}$$

$$H(\pi_k, R_a^l|R_b^l) = -\sum_{\forall \mathbf{x}_{na}^l \in \aleph(\mathbf{x}_{nb}^l)} \sum_{k=1}^{K} P(\pi_k, R_a^l|R_b^l) \log P(\pi_k, R_a^l|R_b^l). \tag{4}$$

The first summation is over all part realizations R_a^l that reside in a neighborhood of all R_b^l such that $\mathbf{x}_{na}^l \in \aleph(\mathbf{x}_{nb}^l)$, for all $i = Y_a^l$ and $j = Y_b^l$, C is a set of cluster ids, $K = |C|$ is the number of clusters, $\pi_k \in C$ is a cluster label, and $P(\pi_k, R_a^l|R_b^l) \triangleq P_{\mathcal{P}_i^l}(\pi_k, R_a^l|\mathcal{P}_j^l = R_b^l)$.

The pairwise statistical relationship between two part realizations R_a^l and R_b^l is represented as $M_{ijk} = (i, j, \mathbf{c}_{ijk}, Z_{ijk})$, where $\mathbf{c}_{ijk}$ is the center position of the k^{th} cluster. In the construction of an object graph $\mathbb{G}_l$ at the l^{th} layer, we compute $\phi_{ab}^l = (\mathbf{c}_{ijk}, \hat{k})$, $\forall a, b$ as $\hat{k} = \arg\min_{k \in C} \|\mathbf{d}_{ab} - \mathbf{c}_{ijk}\|_2$, where $\|\cdot\|_2$ is the Euclidean distance, $i = Y_a^l$ and $j = Y_b^l$, $\mathbf{d}_{ab} = \mathbf{x}_{na} - \mathbf{x}_{nb}$, $\mathbf{x}_{na}$ and $\mathbf{x}_{nb}$ are the positions of R_a^l and R_j^l in an image s_n, respectively.

Inference of Compositions of Parts Using MDL. Given a set of parts $\mathcal{P}^l$, a set of part realizations $\mathcal{R}^l$, and an object graph $\mathbb{G}_l$ at the l^{th} layer, we infer compositions of parts at the $(l+1)^{st}$ layer by computing a mapping $\Psi_{l,l+1}$ in (1). In this mapping, we search for a structure which *best describes* the structure of parts $\mathcal{P}^l$ as the compositions constructed at the $(l+1)^{st}$ layer by minimizing the length of description of $\mathcal{P}^l$. In the inference process, we search a set of graphs $\mathcal{G}^{l+1} = \{\mathcal{G}_j^{l+1}\}_{j=1}^{A_{l+1}}$ which minimizes the description length of $\mathbb{G}_l$ as

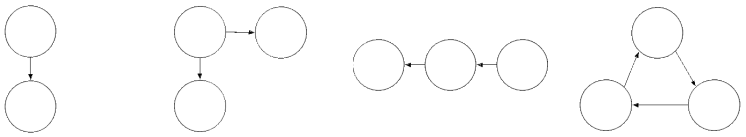
$$\mathcal{G}^{l+1} = \arg\min_{\mathcal{G}_j^{l+1}} value(\mathcal{G}_j^{l+1}, \mathbb{G}_l), \tag{5}$$

where

$$value(\mathcal{G}_j^{l+1}, \mathbb{G}_l) = \frac{DL(\mathcal{G}_j^{l+1}) + DL(\mathbb{G}_l|\mathcal{G}_j^{l+1})}{DL(\mathbb{G}_l)}. \tag{6}$$

is the compression value of an object graph $\mathbb{G}_l$ given a subgraph $\mathcal{G}_j^{l+1}$ of a receptive graph RG_j^l, $\forall j = 1, 2, \ldots, B_l$. Description length DL of a graph G is calculated using the number of bits to represent node labels, edge labels and adjacency matrix, as explained in [3]. The inference process consists of two steps:

1. **Enumeration:** In the graph enumeration step, candidate graphs $\mathcal{G}^{l+1}$ are generated from $\mathbb{G}_l$. However, each $\mathcal{G}_j^{l+1} \in \mathbb{G}_l$ is required to include nodes



(a) $\mathcal{G}_1^{l+1}$ (Valid) **(b)** $\mathcal{G}_2^{l+1}$ (Valid) **(c)** $\mathcal{G}_3^{l+1}$ (Invalid) **(d)** $\mathcal{G}_4^{l+1}$ (Invalid)

Fig. 2. Valid and invalid candidates

Input : $\mathbb{G}_l = (\mathbb{V}_l, \mathbb{E}_l)$: Object graph, *beam*, *numBest*, *bestPartSize*.
Output: Parts $\mathcal{P}^{l+1}$, realizations $\mathcal{R}^{l+1}$.
1 $parentList := null$; $childList := null$; $bestPartList := null$;
where *childList*,*bestPartList* are priority queues ordered by MDL scores.
2 Initialize *parentList* with frequent single node parts;
3 **while** *parentList is not empty* **do**
4 Extend parts in *parentList* in all possible ways into *childList*;
5 Evaluate parts in *childList* using (6);
6 Trim *childList* to *beam* top parts;
7 Merge elements of *childList* and *bestPartList* into *bestPartList*;
8 $parentList := null$;
9 Swap *parentList* and *childList*;
end
10 Trim *bestPartList* to *maxBest* top parts;
11 $\mathcal{P}^{l+1} := bestPartList$;
12 $\mathcal{R}^{l+1} := bestPartList.getInstances()$;

Algorithm 1: Inference of new Compositions

$\mathcal{V}_j^{l+1}$ and edges $\mathcal{E}_j^{l+1}$ from only one receptive graph RG_i^l, $\forall i$. This selective candidate generation procedure enforces $\mathcal{G}_j^{l+1}$ to represent an area around its centre node. Examples of valid and invalid candidates are illustrated in Fig. 2. $\mathcal{G}_1^{l+1}$ and $\mathcal{G}_2^{l+1}$ are valid structures since each graph is inferred from a single receptive graph, e.g. RG_1^l and RG_2^l, respectively. Invalid graphs $\mathcal{G}_3^{l+1}$ and $\mathcal{G}_4^{l+1}$ are not enumerated since their nodes/edges are inferred from multiple receptive graphs.

2. **Evaluation:** Once we obtain $\mathcal{G}^{l+1}$ by solving (5) with $\mathcal{G}^{l+1}$ subject to constraints provided in the previous step, we compute a set of graph instances of part realizations $G^{l+1} = \{G_i^{l+1}\}_{i=1}^{B_{l+1}}$ such that $G_i^{l+1} \in iso(\mathcal{G}_j^{l+1})$ and $G_i^{l+1} \subseteq \mathbb{G}_l$, where $iso(\mathcal{G}_j^{l+1})$ is a set of all subgraphs that are isomorphic to $\mathcal{G}_j^{l+1}$. This problem is defined as a subgraph isomorphism problem [4], which is NP-complete. In this work, the proposed graph structures are acyclic and star-shaped, enabling us to solve (5) in P-time. In order to obtain two sets of subgraphs $\mathcal{G}^{l+1}$ and G^{l+1} by solving (5), we have implemented a simplified version of the substructure discovery system, SUBDUE [4] which is employed in a restricted search space. The discovery algorithm is explained in

Algorithm 1. The key difference between the original SUBDUE and our implementation is that in Step **4**, *childList* contains only star-shaped graphs, which are extended from *parentList* by single nodes. The parameters *beam*, *numBest*, *bestPartSize* are used to prune the search space.

The label of a part $\mathcal{P}_j^{l+1}$ is defined according to its compression value $\mu_j^{l+1} \triangleq value(\mathcal{G}_j^{l+1}, \mathbb{G}_l)$ computed in (6). We sort compression values in ascending order, and assign the part label to the index of the compression value of the part.

After sets of graphs and part labels are obtained at the $(l+1)^{st}$ layer, we construct a set of parts $\mathcal{P}^{l+1} = \{\mathcal{P}_i^{l+1}\}_{i=1}^{A_{l+1}}$, where $\mathcal{P}_i^{l+1} = (\mathcal{G}_i^{l+1}, \mathcal{Y}_i^{l+1})$. We call $\mathcal{P}^{l+1}$ a set of *compositions* of the parts from $\mathcal{P}^l$, constructed at the $(l+1)^{st}$ layer. Similarly, we extract a set of part realizations $\hat{R}^{l+1} = \{R_j^{l+1}\}_{j=1}^{B_{l+1}}$, where $R_j^{l+1} = (G_j^{l+1}, Y_j^{l+1})$. In order to remove the redundancy in $\hat{R}^{l+1}$, we perform local inhibition as in [10] and obtain a new set of part realizations $R^{l+1} \subseteq \hat{R}^{l+1}$.

Incremental Construction of the Vocabulary

Definition 3 (Vocabulary). *A tuple $\Omega_l = (\mathcal{P}^l, \mathcal{M}^l)$ is the vocabulary constructed at the l^{th} layer using the training set S^{tr}. The vocabulary of a CHOP with L layers is defined as the set $\Omega = \{\Omega_l : l = 1, 2, \ldots, L\}$.* □

We construct Ω of CHOP incrementally as described in the pseudo-code of the vocabulary learning algorithm given in Algorithm 2. In the first step of the algorithm, we extract a set of Gabor features $F_n = \{f_{nm}(\mathbf{x}_{nm})\}_{m=1}^{M}$ from each image $s_n \in S^{tr}$ using Gabor filters employed at location $\mathbf{x}_{nm}$ in s_n at Θ orientations. Then, we perform local inhibition of Gabor features using non-maxima suppression to construct a set of suppressed Gabor features $\hat{F}_n \subset F_n$ as described in Section 3.1 in the second step. Next, we initialize the variable l which defines the layer index, and we construct parts $\mathcal{P}^1$ and part realizations R^1 at the first layer as described in Definition 1.

In steps **5 – 11**, we incrementally construct the vocabulary of the CHOP. In step **5**, we compute the sets of modes $\mathcal{M}^l$ by learning statistical relationships between part realizations as described in Section 3.2. In the sixth step, we construct an object graph $\mathbb{G}_l$ using $\mathcal{M}^l$ as explained in Definition 2, and we construct the vocabulary $\Omega_l = (\mathcal{P}^l, \mathcal{M}^l)$ at the l^{th} layer in step **7**. Next, we infer part graphs that will be constructed at the next layer $\mathcal{G}^{l+1}$ by computing the mapping $\Psi_{l,l+1}$. For this purpose, we solve (5) using our graph mining implementation to obtain a set of parts $\mathcal{P}^{l+1}$ and a set of part realizations R^{l+1} as explained in Section 3.2. We increment l in step **10**, and subsample the positions of part realizations R_i^l by a factor of σ, $\forall n, R_i^l$ in step **11**, which effectively increases the area of the receptive fields through upper layers. We iterate the steps **5 – 11** while a non-empty part graph $\mathcal{G}_i^l$ is either obtained from the training images at the first layer, or inferred from Ω_{l-1}, R^{l-1} and $\mathbb{G}_{l-1}$ at $l > 1$, i.e. $\mathcal{G}^l \neq \varnothing$, $\forall l \geq 1$. As the output of the algorithm, we obtain the vocabulary of CHOP, $\Omega = \{\Omega_l : l = 1, 2, \ldots, L\}$.

Input :
- $S^{tr} = \{s_n\}_{n=1}^{N}$: Training dataset,
- Θ: The number of different orientations of Gabor features,
- σ: Subsampling ratio.

Output: Vocabulary Ω.

1 Extract a set of Gabor features $F^{tr} = \bigcup_{n=1}^{N} F_n^{tr}$, where $F_n^{tr} = \{f_{nm}(\mathbf{x}_{nm})\}_{m=1}^{M}$ from each image $s_n \in S^{tr}$;
2 Construct a set of suppressed Gabor features $\hat{F}^{tr} \subset F^{tr}$ (see Section 3.1);
3 $l := 1$;
4 Construct $\mathcal{P}^1$ and R^1 (see Definition 1);
while $\mathcal{G}^l \neq \varnothing$ **do**
5 Compute the sets of modes $\mathcal{M}^l$ (see Section 3.2);
6 Construct $\mathbb{G}_l$ using $\mathcal{M}^l$ (see Definition 2);
7 Construct $\Omega_l = (\mathcal{P}^l, \mathcal{M}^l)$;
8 Infer part graphs $\mathcal{G}^{l+1}$ by solving (5) (see Section 3.2);
9 Construct $\mathcal{P}^{l+1}$ and R^{l+1} (see Section 3.2);
10 $l := l + 1$;
11 Subsample the positions of part realizations R_i^l by a factor of σ, $\forall n, R_i^l$;
end
12 $\Omega = \{\Omega_t : t = 1, 2, \ldots, l-1\}$;

Algorithm 2: The vocabulary learning algorithm of Compositional Hierarchy of Parts

3.3 Inference of Object Shapes on Test Images

In the testing phase, we infer shapes of objects on test images $s_n \in S^{te}$ using the learned vocabulary of parts Ω. We incrementally construct a set of inference graphs $\mathcal{T}(s_n) = \{\mathcal{T}_l(s_n)\}_{l=1}^{L}$ of a given test image $s_n \in S^{te}$ using the learned vocabulary $\Omega = \{\Omega_l\}_{l=1}^{L}$. At each l^{th} layer, we construct a set of part realizations $R^l(s_n) = \left\{R_i^l(s_n) = \left(G_i^l(s_n), Y_i^l(s_n)\right)\right\}_{i=1}^{B'_l}$ and an object graph $\mathbb{G}_l = (\mathbb{V}_l, \mathbb{E}_l)$ of s_n, $\forall l = 1, 2, \ldots, L$. Algorithm 3 explains the inference algorithm for test images. The test image is processed in the same manner as in vocabulary learning (steps **1 – 5**). In step **6**, isomorphisms of part graph descriptions $\mathcal{G}^{l+1}$ obtained from Ω_{l+1} are searched in $\mathbb{G}_l$ in P-time (see Section 3.2). Part realizations R^{l+1} of the new object graph $\mathbb{G}_{l+1}$ are extracted from G^{l+1} in step **7**. The discovery process continues until no new realizations are found.

At the first layer $l = 1$, the nodes of the instance graph $G_i^1(s_n)$ of a part realization $R_i^1(s_n)$ represent the Gabor features $f_{na}^i(\mathbf{x}_{na}) \in \hat{F}_n^{te}$ observed in the image $s_n \in S^{te}$ at an image location $\mathbf{x}_{na}$ as described in Section 3.2. In order to infer the graph instances and compositions of part realizations in the following layers $1 < l \leq L$, we employ a graph matching algorithm that constructs $G_i^{l+1}(s_n) = \{H(\mathcal{P}^{l+1}) : H(\mathcal{P}^{l+1}) \subseteq \mathbb{G}_l\}$ which is a set of subgraph isomorphisms $H(\mathcal{P}^{l+1})$ of part graphs $\mathcal{G}^{l+1}$ in $\mathcal{P}^{l+1}$, computed in $\mathbb{G}_l$.

Input :
- s: Test image,
- Ω: Vocabulary,
- Θ: The number of different orientations of Gabor features,
- σ: Subsampling ratio.

Output: Inference graph $\mathcal{T}(s)$.

Extract a set of Gabor features $F = \{f_m(\mathbf{x}_m)\}_{m=1}^{M}$ from image s;
Construct a set of suppressed Gabor features $\hat{F} \subset F$ (see Section 3.1);
$l := 1$;
Construct R^1 from $\hat{F}$ (see Definition 1);
while $\Omega_{l+1} \neq \varnothing \wedge R^l \neq \varnothing$ **do**
    Construct $\mathbb{G}_l$ using $\mathcal{M}^l$ in Ω_l;
    Find graph instances of part realizations $G^{l+1} = \{G_j^{l+1}\}_{j=1}^{B'^{l+1}}$ such that $G_j^{l+1} \in iso(\mathcal{G}^{l+1})$ and $G_j^{l+1} \subseteq \mathbb{G}_l$ (see Section 3.2, **Evaluation**);
    Construct R^{l+1} from G^{l+1} (see Section 3.2);
    $l := l + 1$;
    Subsample the positions of part realizations R_i^l by a factor of σ, $\forall R_i^l$;
end
$\mathcal{T}(s) = \{\mathbb{G}_t : t = 1, 2, \ldots, l-1\}$;

Algorithm 3: Object shape inference algorithm for test images

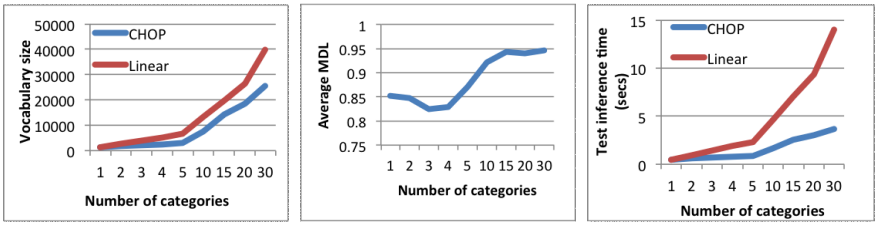
4 Experiments

We examine our proposed approach and algorithms on six benchmark object shape datasets, which are namely the Washington image dataset (Washington) [1], the MPEG-7 Core Experiment CE-Shape 1 dataset [14], the ETHZ Shape Classes dataset [9], 40 sample articulated Tools dataset (Tools-40) [17], 35 sample multi-class Tools dataset (Tools-35) [2] and the Myth dataset [2]. In the experiments, we used $\Theta = 6$ different orientations of Gabor features with the same Gabor kernel parameters implemented in [10]. We used a subsampling ratio of $\sigma = 0.5$. A Matlab implementation of CHOP is available here[1]. Additional analyses related to part shareability and qualitative results are given in the Supplementary Material.

4.1 Analysis of Generative and Descriptive Properties

We analyze the relationship between the number of classes, views, objects, and vocabulary size, average MDL values and test inference time in three different setups, respectively. Vocabulary size and test inference time analyses provide information about the part shareability and generative shape representation behavior of our algorithm. We examine the variations of the average MDL values under different test sets. In order to get a more descriptive estimate of MDL values, we use 10 best parts constructed at each layer of CHOP. While a vocabulary

[1] `https://github.com/rusen/CHOP.git`



(a) **(b)** **(c)**

Fig. 3. Analyses with different number of categories. (Best viewed in colour).

layer may contain thousands of parts, most of the parts constructed with the lowest MDL scores belong to a single object in the model, and therefore exhibit no shareability.

The inference time of CHOP is calculated by averaging running times of the inference algorithm which is employed on test images.

Analyses with Different Number of Categories. In this section we use the first 30 categories of the MPEG-7 Core Experiment CE-Shape 1 dataset [14]. We randomly select 5 images from each category to construct training sets.

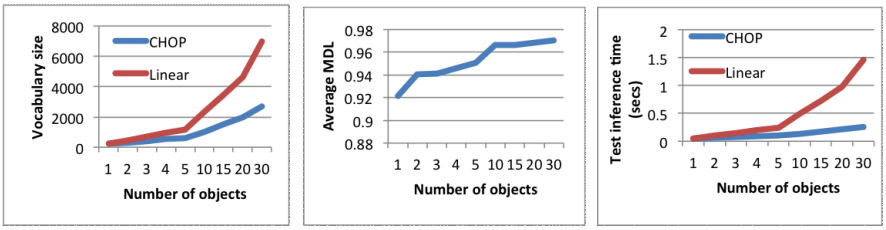
The vocabulary size grows sub-linearly as shown with the blue line in Fig. 3.a. The higher part shareability observed in the first layers of CHOP is considered as the main contributing factor which affects the vocabulary size. We observe a sub-linear growth of the number of parts as the number of categories increases, which affects the test image inference time as shown in Fig. 3.c. This is observed because the inference process requires searching every composition in the vocabulary within the graph representation of a test image. The efficient indexing mechanism implemented in CHOP speeds up the testing time, and the average test time is calculated as 0.5-3 seconds depending on the number of categories. Average MDL values tend to increase after a boost at around 3-4 categories (lower is better), and converge at 15 categories. The inter-class appearance differences allow for a limited amount of shareability between categories.

4.2 Analyses with Different Number of Objects

In order to analyze the effect of increasing number of images to the proposed performance measures, we use 30 samples belonging to the "Apple Logos" class in ETHZ Shape Classes dataset [9] for training. Compared to the results obtained in the previous section, we observe that average MDL values increase gradually as the number of objects increase in Fig. 4.b. Additionally, the growth rate of the vocabulary size observed in Fig. 4.a is less than the one depicted in Fig. 3.a.

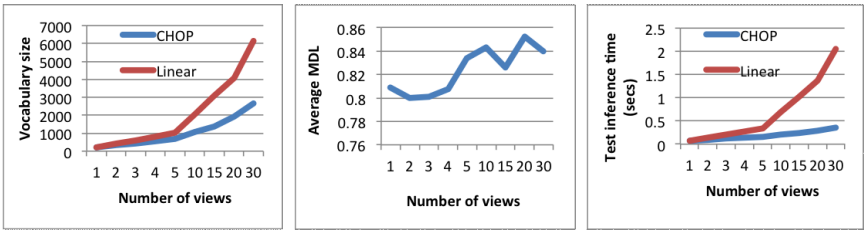
4.3 Analyses with Different Number of Views

In the third set of experiments, we use a subset of Washington image dataset [1] consisting of images captured at different views of the same object. Multiple



(a) **(b)** **(c)**

Fig. 4. Analyses with different number of objects. (Best viewed in colour).



(a) **(b)** **(c)**

Fig. 5. Analyses with different number of views. (Best viewed in colour).

view images of a cup are used as the training data. Due to the fairly symmetrical nature of a cup except for its textures and handle, the shareability of the parts in the vocabulary remains consistent as the training image set grows. Interestingly, we observe a local maximum at around 15 views in Fig. 5.b. Depending on the inhibition and part selection (SUBDUE) parameters, less frequently observed yet valuable parts may be discarded by the algorithm in mid-layers.

4.4 Shape Retrieval Experiments

Following the results of [20,22,23], we employ eigenvalues of adjacency matrices of edge weighted graphs computed using object graphs of shapes as shape descriptors. For this purpose, we first define edge weights $e_{ab} \in E_l$ of an edge weighted graph $W_l = (V_l, E_l)$ of an object graph $\mathbb{G}_l = (\mathbb{V}_l, \mathbb{E}_l)$ as

$$e_{ab} = \begin{cases} \pi_k, & \text{if} \quad \mathrm{R}^{\mathrm{l}}_{\mathrm{a}} \text{ is connected to } \mathrm{R}^{\mathrm{l}}_{\mathrm{b}}, \quad \forall R^l_a, R^l_b \in \mathbb{V}_l \\ 0, & otherwise \end{cases}, \tag{7}$$

where π_k is the cluster index which minimizes the conditional entropy (4) in (3). Then, we compute the weighted adjacency matrix of W_l and use the eigenvalues as shape descriptors. We compute the distance between two shapes as the Euclidean distance between their shape descriptors.

In the first set of experiments, we compare the retrieval performances of CHOP and the state-of-the-art shape classification algorithms which use inner-distance (ID) measures to compute shape descriptors which are robust to articulation [17]. The experiments are performed on Tools-40 dataset [17] which

Table 1. Comparison of shape retrieval performances (%) on Tools-40 dataset

Algorithms	Top 1	Top 2	Top 3	Top 4
SC+DP [17]	20/40	10/40	11/40	5/40
MDS+SC+DP [17]	36/40	26/40	17/40	15/40
IDSC+DP [17]	40/40	34/40	35/40	27/40
CHOP	37/40	35/40	35/40	29/40

contains 40 images captured using 8 different objects each of which provides 5 articulated shapes. Given each query image, the four most similar matches are chosen from the other images in the dataset for the evaluation of the recognition results [17]. The results are summarized as the number of first, second, third and fourth most similar matches that come from the correct object in Table 1. We observe that CHOP provides better performance than the shape-based descriptors and retrieval algorithms SC+DP and MDS+SC+DP [17]. However, IDSC+DP [17], which integrates texture information with the shape information, provides better performance for Top 1 retrieval results, and CHOP performs better than IDSC+DP for Top 4 retrieval results. The reason of this observation is that texture of shape structures provides discriminative information about shape categories. Therefore, the objects which have the most similar textures are closer to each other than the other objects as observed in Top 1 retrieval results. On the other hand, texture information may dominate the shape information and may lead to overfitting as observed in Top 4 retrieval results (see Table 1).

In the second set of experiments, we use Myth and Tools-35 datasets in order to analyze the performance of the shape retrieval algorithms [18] and CHOP, considering part shareability and category-wise articulation. In the Myth dataset, there are three categories, namely Centaur, Horse and Man, and 5 different images belonging to 5 different objects in each category. Shapes observed in images differ by articulation and additional parts, e.g. the shapes of objects belonging to Centaur and Man categories share the upper part of the man body, and the shapes of objects belonging to Centaur and Horse categories share the lower part of the horse body. In the Tools-35 dataset, there are 35 shapes belonging to 4 categories which are split as 10 scissors, 15 pliers, 5 pincers, 5 knives. Each object belonging to a category differs by an articulation. Performance values are calculated using a Bullseye test as suggested in [18] to compare the performances of CHOP and other shape retrieval algorithms Contour-ID [18] and Contour-HF [18]. In the Bullseye test, five most similar candidates for each query image are considered [18]. Experimental results given in Table 2 show that CHOP outperforms Contour-ID and Contour-HF [18] which employ distributions of descriptor values calculated at shape contours as shape features that are invariant to articulations and deformations in local part structures. However, part shareability and articulation properties of shapes may provide discriminative information about shape structures, especially on the images in the Myth dataset.

Table 2. Comparison of shape retrieval performances (%) on Myth and Tools-35

Datasets	Contour-ID [18]	Contour-HF [18]	CHOP
Tools-35	84.57	84.57	87.86
Myth	77.33	90.67	93.33

5 Conclusion

We have proposed a graph theoretic approach for object shape representation in a hierarchical compositional architecture called Compositional Hierarchy of Parts (CHOP). Two information theoretic algorithms are used for learning a vocabulary of compositional parts employing a hybrid generative-descriptive model. First, statistical relationships between parts are learned using the MCEC algorithm. Then, part selection problem is defined as a frequent subgraph discovery problem, and solved using an MDL principle. Part compositions are inferred considering both learned statistical relationships between parts and their description lengths at each layer of CHOP.

The proposed approach and algorithms are examined using six benchmark shape datasets consisting of different images of an object captured at different viewpoints, and images of objects belonging to different categories. The results show that CHOP can use part shareability property in the construction of *compact* vocabularies and inference trees efficiently. For instance, we observe that the running time of CHOP to perform inference on test images is approximately 0.5-3 seconds for an image. Additionally, we can construct compositional shape representations which provide part realizations that completely cover the shapes on the images. Finally, we compared shape retrieval performances of CHOP and the state-of-the-art retrieval algorithms on three benchmark datasets. The results show that CHOP outperforms the evaluated algorithms using part shareability and fast inference of descriptive part compositions.

In the future work, we will employ discriminative learning for pose estimation and categorization of shapes. In addition, online and incremental learning will be implemented considering the results obtained from the analyses on part shareability performed in this work.

Acknowledgement. This work was supported in part by the European Commission project PaCMan EU FP7-ICT, 600918. The authors would also like to thank Sebastian Zurek for helpful discussions.

References

1. Bo, L., Lai, K., Ren, X., Fox, D.: Object recognition with hierarchical kernel descriptors. In: Proceedings of the 2011 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2011, pp. 1729–1736. IEEE Computer Society, Washington, DC (2011)

2. Bronstein, A.M., Bronstein, M.M., Bruckstein, A.M., Kimmel, R.: Analysis of two-dimensional non-rigid shapes. Int. J. Comput. Vision 78(1), 67–88 (2008)
3. Cook, D.J., Holder, L.B.: Substructure discovery using minimum description length and background knowledge. J. Artif. Int. Res. 1(1), 231–255 (1994), http://dl.acm.org/citation.cfm?id=1618595.1618605
4. Cook, D.J., Holder, L.B.: Mining Graph Data. John Wiley & Sons (2006)
5. Davies, R.H., Twining, C.J., Cootes, T.F., Waterton, J.C., Taylor, C.J.: A minimum description length approach to statistical shape modeling. IEEE Trans. Med. Imag. 21(5), 525–537 (2002)
6. Delong, A., Gorelick, L., Veksler, O., Boykov, Y.: Minimizing energies with hierarchical costs. Int. J. Comput. Vision 100(1), 38–58 (2012)
7. Felzenszwalb, P., Girshick, R., McAllester, D., Ramanan, D.: Object detection with discriminatively trained part-based models. IEEE Trans. Pattern Anal. Mach. Intell. 32(9), 1627–1645 (2010)
8. Felzenszwalb, P., Schwartz, J.: Hierarchical matching of deformable shapes. In: Proceedings of the 2007 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 1–8 (June 2007)
9. Ferrari, V., Tuytelaars, T., Van Gool, L.: Object detection by contour segment networks. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) ECCV 2006. LNCS, vol. 3953, pp. 14–28. Springer, Heidelberg (2006)
10. Fidler, S., Leonardis, A.: Towards scalable representations of object categories: Learning a hierarchy of parts. In: Proceedings of the 2007 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2007, pp. 1–8 (June 2007)
11. Fidler, S., Boben, M., Leonardis, A.: Learning hierarchical compositional representations of object structure. In: Dickinson, S.J., Leonardis, A., Schiele, B., Tarr, M.J. (eds.) Object categorization computer and human perspectives, pp. 196–215. Cambridge University Press, Cambridge (2009)
12. Guo, C.-E., Zhu, S.-C., Wu, Y.N.: Modeling visual patterns by integrating descriptive and generative methods. Int. J. Comput. Vision 53(1), 5–29 (2003)
13. Kokkinos, I., Yuille, A.: Inference and learning with hierarchical shape models. Int. J. Comput. Vis. 93(2), 201–225 (2011)
14. Latecki, L., Lakamper, R., Eckhardt, T.: Shape descriptors for non-rigid shapes with a single closed contour. In: Proceedings of the 2000 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 1, pp. 424–429 (June 2000)
15. Levinshtein, A., Sminchisescu, C., Dickinson, S.J.: Learning hierarchical shape models from examples. In: Rangarajan, A., Vemuri, B.C., Yuille, A.L. (eds.) EMMCVPR 2005. LNCS, vol. 3757, pp. 251–267. Springer, Heidelberg (2005)
16. Li, H., Zhang, K., Jiang, T.: Minimum entropy clustering and applications to gene expression analysis. In: Proceedings of the 2004 IEEE Computational Systems Bioinformatics Conference, CSB 2004, pp. 142–151. IEEE Computer Society, Washington, DC (2004)
17. Ling, H., Jacobs, D.: Shape classification using the inner-distance. IEEE Trans. Pattern Anal. Mach. Intell. 29(2), 286–299 (2007)
18. Nanni, L., Brahnam, S., Lumini, A.: Local phase quantization descriptor for improving shape retrieval/classification. Pattern Recogn. Lett. 33(16), 2254–2260 (2012)
19. Ommer, B., Buhmann, J.M.: Learning the compositional nature of visual object categories for recognition. IEEE Trans. Pattern Anal. Mach. Intell. 32(3), 501–516 (2010)

20. Raviv, D., Kimmel, R., Bruckstein, A.M.: Graph isomorphisms and automorphisms via spectral signatures. IEEE Trans. Pattern Anal. Mach. Intell. 35(8), 1985–1993 (2013)
21. Salakhutdinov, R., Tenenbaum, J.B., Torralba, A.: Learning with hierarchical-deep models. IEEE Trans. Pattern Anal. Mach. Intell. 35(8), 1958–1971 (2013)
22. Shokoufandeh, A., Macrini, D., Dickinson, S., Siddiqi, K., Zucker, S.W.: Indexing hierarchical structures using graph spectra. IEEE Trans. Pattern Anal. Mach. Intell. 27(7), 1125–1140 (2005)
23. Siddiqi, K., Shokoufandeh, A., Dickinson, S.J., Zucker, S.W.: Shock graphs and shape matching. Int. J. Comput. Vision 35(1), 13–32 (1999)
24. Torsello, A., Hancock, E.R.: Learning shape-classes using a mixture of tree-unions. IEEE Trans. Pattern Anal. Mach. Intell. 28(6), 954–967 (2006)
25. Zhu, A.L., Chen, Y., Yuille: Learning a hierarchical deformable template for rapid deformable object parsing. IEEE Trans. Pattern Anal. Mach. Intell. 32(6), 1029–1043 (2010)
26. Zhu, L.L., Chen, Y., Yuille, A.: Recursive compositional models for vision: Description and review of recent work. J. Math. Imaging Vis. 41(1-2), 122–146 (2011)