

Dissimilarity Between Two Skeletal Trees In A Context

E. Baseski^{a,1}, A. Erdem^{b,2}, S. Tari^{b,3,*},

^a*The Maersk MC-Kinney Moller Institute, University of Southern Denmark,
Odense, Denmark*

^b*Department of Computer Engineering, Middle East Technical University,
TR-06531, Ankara, Turkey*

Abstract

Skeletal trees are commonly used in order to express geometric properties of the shape. Accordingly, tree edit distance is used to compute a dissimilarity between two given shapes. We present a new tree edit based shape matching method which uses a recent coarse skeleton representation. The coarse skeleton representation allows us to represent both shapes and shape categories in the form of depth-1 trees. Consequently, we can easily integrate the influence of the categories into shape dissimilarity measurements. The new dissimilarity measure gives a better within group versus between group separation, and it mimics the asymmetric nature of human similarity judgements.

Key words: skeletal shape matching, shape similarity, disconnected skeleton

1 Introduction

There is a growing interest within both pattern recognition and computer vision communities on characterizing shapes by their relations to other shapes rather than by using features or templates, *e.g.* [6–8, 13, 14, 25, 26, 31]. In these works, the concept of (dis)similarity takes the center stage. Dissimilarity can be related to some shape distance in some metric space and may be used for retrieval from shape databases.

Typically, the geometric similarity between two shapes is a measure of how well the primitives forming the shapes and/or their spatial organizations agree [10, 19, 21, 23, 32, 34, 38, 39, 41, 44, 46]. Tree data structure has been widely used for describing shapes, as it provides a natural representation of the inclusion relations of the primitives. When a shape (primitives and their inclusion relations) is represented by a tree, the best correspondence between two given shapes can be expressed as the best partial match between their trees. Accordingly, the shape dissimilarity is computed as the edit distance which is defined as the cost of transforming the first tree into the second one by using node removal, node insertion and attribute change operations [48]. In the shape literature, it is an accepted practice to form tree or graph descriptions using shape skeletons, and to match these descriptions using edit distance [12, 23, 32, 38, 39]. Typically, these works are generic and they ignore contextual effects, despite the observation that human dissimilarity judgements are biased by the other

* Corresponding author.

Email addresses: `emre@mip.sdu.dk` (E. Baseski), `aykut@ceng.metu.edu.tr` (A. Erdem), `stari@metu.edu.tr` (S. Tari).

¹ This work was completed when E. Baseski was at METU.

² Supported in part by TUBITAK-BAYG PhD scholarship.

³ Supported in part by the research grant TUBITAK-105E154.

shapes [14, 17, 25, 31, 35, 49]. In applied problems, e.g. shape retrieval from a database, compatibility of the similarity measure with human judgements is desired [36]. How similar things appear depends on whether they are in the same or in the different categories. As an example, consider the following experiment. Place a pair of eyeglasses and an eyeglass case on an empty desk. How similar are these two objects? Add a pair of sunglasses and judge the similarity again. Then add a pencil and a pencil holder. Try to judge the similarity between the first two objects by varying the other objects that you add.

Motivated by the importance of context in the similarity judgements and the widespread use of the skeletal edit operations in shape matching, we propose a new skeletal tree based representation in which contextual effects are incorporated. Following the definition in [50], we interpret the context as any information relevant to the dissimilarity computation task without being directly related to the geometric properties of the compared shapes. In the literature, the context, typically refers to a collection of neighboring entities *e.g.* neighboring objects for a given object [17, 25], local neighborhood of a given pixel [40], or prior knowledge and expectations [24]. The context, in our work, refers to a collection of neighboring shapes or shapes of the same category. We define the dissimilarity between two shapes by respecting the asymmetric roles played by the query shape and the database shape (memory benchmark) [14, 25]. That is, a query shape A is compared to a database shape B whose category is known. Shapes in the category of B form a context which biases dissimilarity computation by modifying the importance of primitives and distances between attributes, as in the philosophy of recent works such as [27, 30].

An interesting feature of our method is that we use rooted-depth-1 tree as the main data structure to represent both individual shapes and categories. We are able to parse shapes into depth-1 trees in a stable and reliable way with the help of Aslan’s skeletal representation [1–3]. Aslan skeleton is extracted from a special surface whose level curves are extensively smoothed versions of the initial shape boundary (Fig. 1 (a)). This surface has a single extremum point catching the center of a blob-like representation of the shape. The skeleton branches (Fig. 1 (b)) can be extracted and classified as positive or negative by using the method in [42]. Positive branches are analogous to the Blum skeleton [9] and they emanate from a positive curvature maxima of the boundary (protrusion). Negative branches emanate from a negative curvature minima (indentation) or a positive curvature minima, and they resemble Leyton’s PISA [22]. Unlike PISA branches, both types of branches in Aslan skeleton grow inside the shape towards the shape center. If the symmetry at the center is n -fold, there are n positive and n negative branches that meet at the shape center. These type of branches are called *major branches*. The remaining branches terminate at a *disconnection point*. Disconnection points are the locations where a positive branch and a negative branch meet. Aslan’s skeletal representation trades detail with the numerical stability and provides a description at the coarsest scale which permits skeletal analysis.

In our representation, each primitive is the disconnection point [1–3] of a simple skeletal branch. Note that a skeleton is a shrinkage of a shape boundary. In this sense, each point on a positive (negative) branch explains two locally symmetric boundary points. At the point of disconnection, the skeleton branch collapses into a single point which explains a segment on the boundary. This segment is bounded by two negative (positive) branches neighboring the pos-

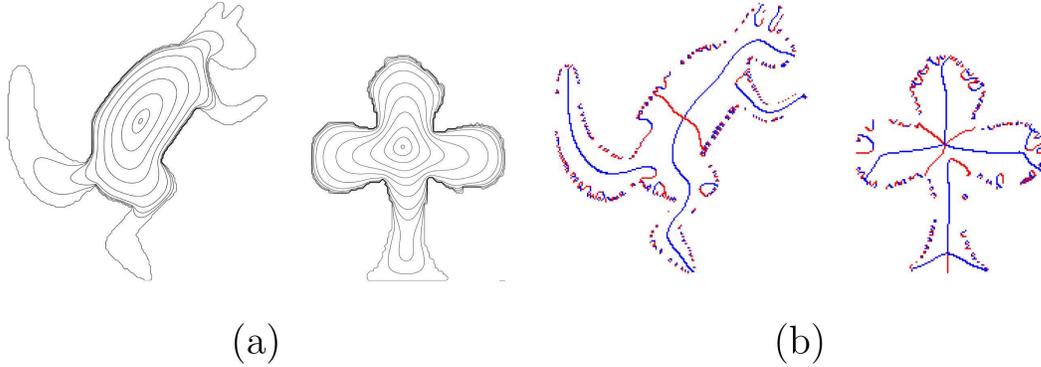


Fig. 1. The method of Aslan. (a) Level curves of the Aslan surface. (b) Skeleton branches. Notice that each positive branch merges with a negative branch at the branch termination or disconnection point. Positive skeleton branches are shown in blue whereas the negative ones are shown in red.

itive (negative) branch. Therefore, we can imagine unfolding each branch to a segment of the shape boundary. These segments are equivalent to the primitive sequences of Nishida [28,29] or to the contour fragments of Grandidier, Saborin and Suen [18]. This is clearly visible in a later figure (Fig. 6). Many developments by Nishida [28,29] such as primitive generation by continuously transforming the collection of boundary fragments can be adopted to enrich our method. Rather than modifying the importance of contour fragments as in [18], we modify the importance of disconnection points in a context.

An interesting argument in favor of context dependence in pattern recognition comes from the Ugly Duckling Theorem [49] which asserts that in the absence of any bias, any two shapes are equally similar. An important implication of the theorem is that there are no privileged primitives. One idea is to start with many primitives each of which provides a rough representation, and then to select the best ones in a given context [33,47]. Our approach is in the opposite direction. We start with a very coarse scale description in which only the numerically stable primitives are kept. The scale is automatically determined by the shape and it is the coarsest scale which permits skeletal analysis [2].

The context utilized in a certain shape comparison is the category tree of the database shape. A category tree is a depth-1 tree which collectively holds all the shapes in that category. Forming a union of tree representations has been previously addressed by Torsello and Hancock [43]. Unlike Torsello and Hancock’s construction, our construction naturally produces a tree which is also depth-1. Representing both individual shapes and categories using the same data structure makes the necessary constructions and computations trivial and allows us to define a flexible mechanism for updating categories as new shapes are observed and categorized. Furthermore, the depth-1 nature of our trees reduces tree edit computations to string-edit computations which we implement by adopting an algorithm of Zhang and Shasha [51].

We test our constructions on a shape retrieval problem by performing two separate experiments. In the first experiment, we compute dissimilarities between every pair in the database, and retrieve the closest shapes to any given shape. In very large shape databases, a single retrieval task becomes computationally intractable when an exhaustive search strategy is used, *i.e.* by comparing the query shape with all the database shapes. Comparing the query shape to a group of shapes, at once, can speed up the process. Accordingly, in the second experiment, we exploit structural equivalence of the shape and the category trees, in order to use our constructions for comparing a shape to a category. Experiments are conducted on two different data sets consisting of 180 and 1000 shapes, respectively. The first data set (Fig. 2) is identical to the one used in [1, 2], in order to facilitate a comparison. The second data set (Fig. 3) is formed by extending the first set with shapes collected from various sources [21, 37].

Our results demonstrate both qualitative and quantitative improvements in

dissimilarity measurement and shape matching. The new measure gives better within group versus between group separation and it reflects human dissimilarity judgements better.

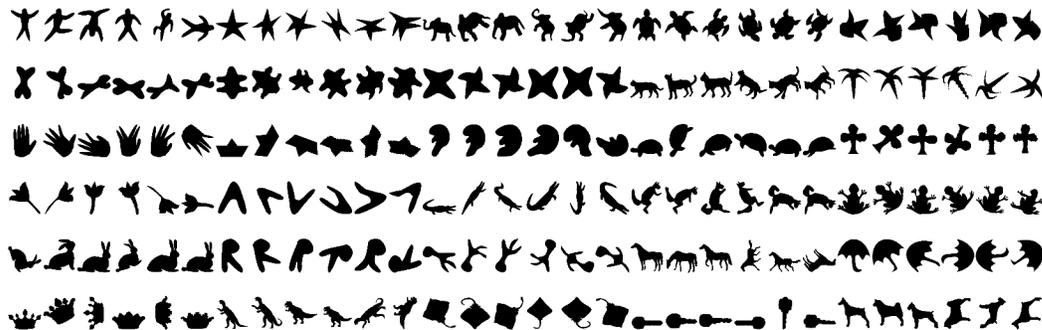


Fig. 2. The small shape database used in the experiments. It contains 180 highly varying shapes from 30 categories.

The paper is organized as follows. Section 2 is on parsing skeleton descriptions into depth-1 trees. In Section 3, the process of matching two given shape trees using edit distance is described. In Section 4, the limitations of the constructions developed in Section 3 are discussed, in order to motivate context guided matching. In Section 5, the constructions for forming a category tree and for comparing two given shapes in a given context are developed. Finally, in Section 6, results are presented and discussed.

2 From Aslan Skeleton to Shape Tree

In Aslan skeleton, relative organization of disconnected skeleton branches can be captured by the location of their disconnection points. The termination concept is artificially extended to the major positive branches(Fig. 4(a)). Disconnection points can be expressed with reference to a shape dependent global coordinate frame that is constructed from the negative local symmetry branches



Fig. 3. The extended shape database used in the experiments. It contains 1000 shapes from 50 categories.

meeting at the shape center (Fig. 4(b)). Alternative coordinate frame constructions are possible depending on the choice of major branch. Quantifiable properties of the disconnected axes such as the termination location (r, θ) and the length l are measured in the same shape dependent coordinate system. These properties serve as attributes.

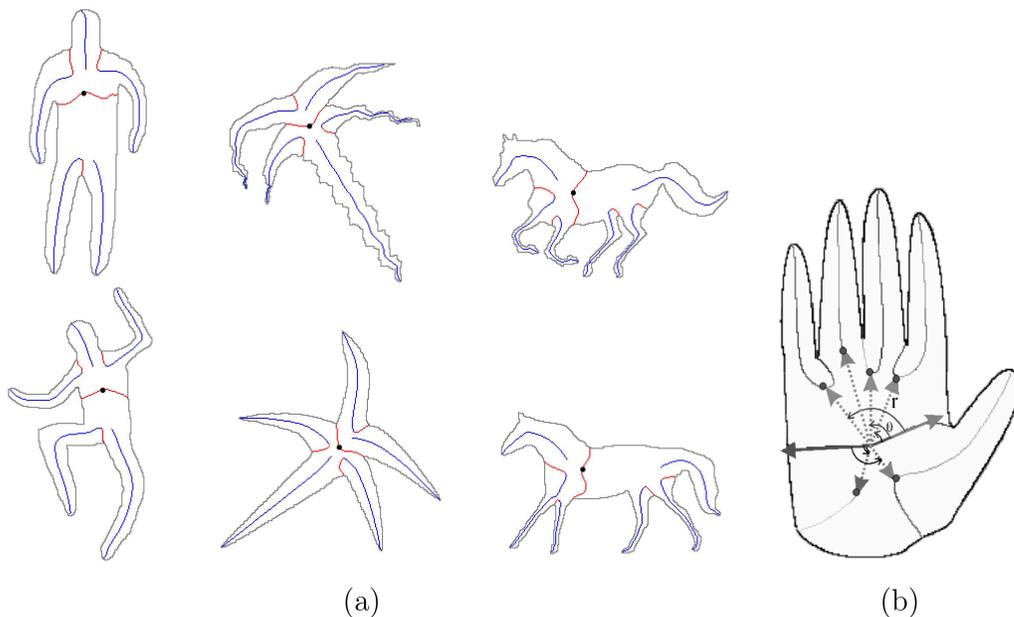


Fig. 4. (a) Sample shapes and their skeletons. Notice that disconnection locations and branch lengths are robust with respect to articulations. (b) Spatial organization of the disconnected skeleton branches (taken from [3]).

Aslan skeletons can be expressed as a tree straightforwardly. For each shape description obtained by using an alternative coordinate frame, we construct a rooted-depth-1 tree (Fig. 5). The root of the tree holds necessary and sufficient information to construct the coordinate system. This information includes the location of the center and the direction and the length normalization factor (based on total branch length). The leaf nodes correspond to the skeleton branches and they hold

- the location of a termination point in polar coordinates (r, θ) ;
- the normalized length l ;

- the branch type as negative or positive.

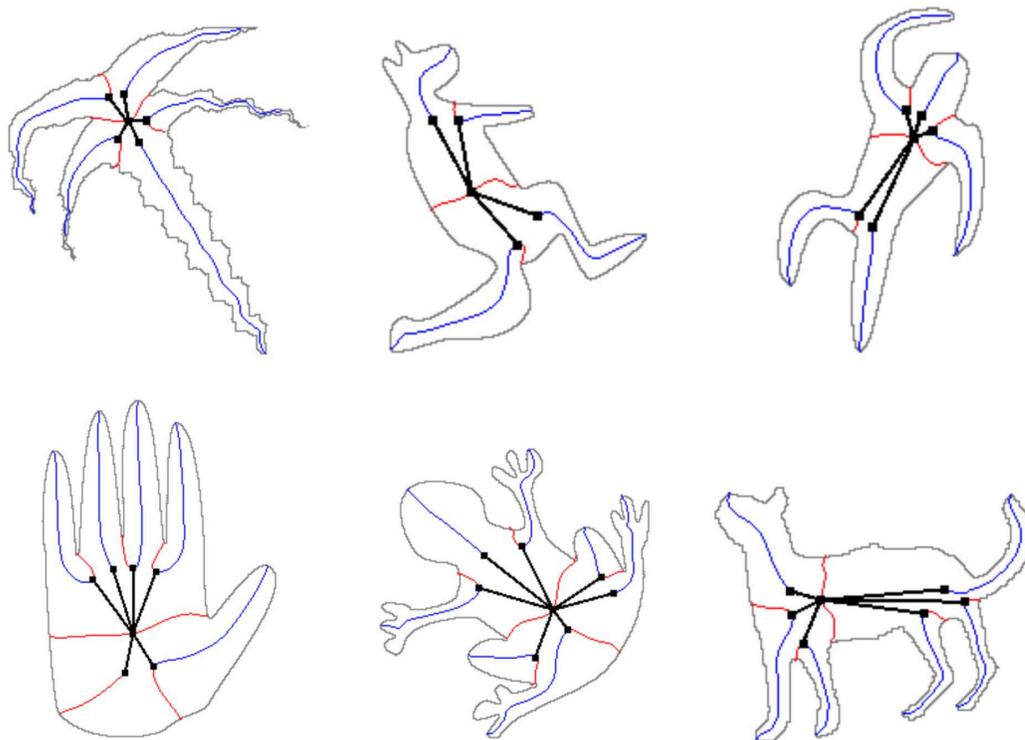


Fig. 5. Sample shape trees. We remark that each disconnection point (except the ones due to major branches) gives rise to two separate nodes in the tree. However for illustration purposes, only one node is drawn.

We remark that branches come in pairs each (one positive and one negative) and a single disconnection point is a termination point for two branches. We include both branches as separate nodes. There may be two exceptions. First, the disconnection point for major branch is an artificially defined point since major branches do not terminate. Second, an extremely short branch may get pruned leaving the branch of opposite type unpaired. The nodes are labeled according to an ordering of branches in order to perform matching in a more efficient way. The ordering may be started from either one of the major negative branches. Thus, we form multiple descriptions of a shape for each such possible choice (Fig. 6). Note that for a shape having n -fold symmetry, there are n possible major negative branches [4].

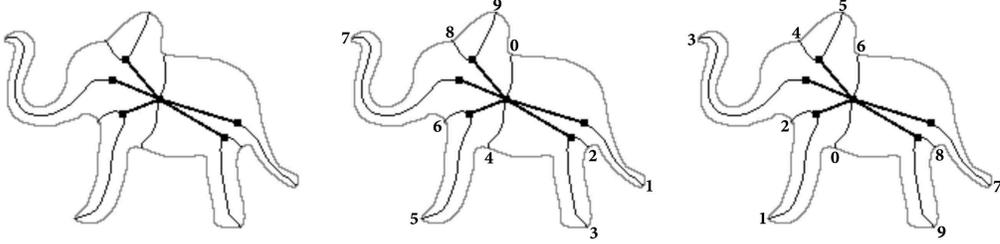


Fig. 6. Multiple descriptions are obtained using two different orderings of the skeleton branches. Notice that the symmetry is two-fold at the center.

3 Matching Shapes with Tree Edit

We define the dissimilarity between two shapes as the minimum cost of transforming one shape tree into another using tree edit operations [51]. The focus of this section is to define editing costs which depend on nodal attributes.

Let \mathcal{T}_1 and \mathcal{T}_2 denote the shape trees to be matched. Since both \mathcal{T}_1 and \mathcal{T}_2 are rooted-ordered-depth-1 trees, each of them can be expressed as a list of nodes (excluding the root):

$$\begin{aligned} \mathcal{T}_1 &= \{u_i = (u_i^r, u_i^\theta, u_i^l, u_i^{type}) \mid u_i \in \mathcal{N}_1\} \\ \mathcal{T}_2 &= \{v_j = (v_j^r, v_j^\theta, v_j^l, v_j^{type}) \mid v_j \in \mathcal{N}_2\} \end{aligned} \quad (1)$$

where i, j denote the order of nodes, (r, θ) is the normalized location of the disconnection point in polar coordinates, $type$ denotes the branch type and l is the normalized length of the corresponding skeleton branch. \mathcal{N}_1 and \mathcal{N}_2 are the set of leaf nodes of \mathcal{T}_1 and \mathcal{T}_2 , respectively.

The edit operations are **rem(ove)**, **ins(ert)** and **ch(ange)**. Let Λ denote the set of nodes removed from \mathcal{T}_1 , Δ denote the set of nodes inserted to \mathcal{T}_1 from \mathcal{T}_2 and Ω denote the set of matched nodes. The matching cost is the minimum

cost of a sequence of edit operations \mathcal{S} :

$$d(\mathcal{T}_1, \mathcal{T}_2) = \min_{\mathcal{S}} \left[\sum_{u \in \Lambda} \mathbf{rem}(u) + \sum_{v \in \Delta} \mathbf{ins}(v) + \sum_{(u,v) \in \Omega} \mathbf{ch}(u,v) \right] \quad (2)$$

The cost functions are described below. Each of them returns a value in the range $[0, 1]$.

- **rem.** This operation removes a node from \mathcal{T}_1 . The corresponding cost function quantitatively measures how well the removed skeleton branch characterizes the shape. Disconnection location of a branch is quite a good measure of significance. While the major branches do not terminate and reach to the shape center, boundary details terminate quite early. As argued in [1–3] disconnected branch length is a good measure of significance. Thus, we define the cost of removing a given node u of \mathcal{T}_1 as follows,

$$\mathbf{rem}(u) = \left(\frac{u^l}{l_{max}(\mathcal{T}_1)} \right) (1 - u^r) \quad (3)$$

where u^l is the length of the branch, u^r is the distance from the shape center and $l_{max}(\mathcal{T}_1)$ is the length of the longest branch of \mathcal{T}_1 . See Fig. 7.

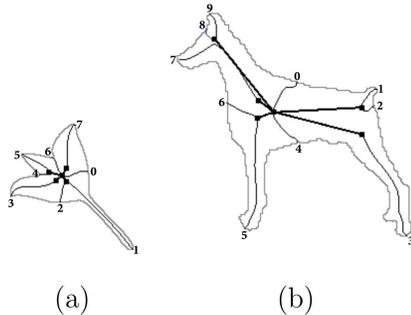


Fig. 7. **rem** cost function on two examples. (a) Since $u_1^l \geq u_5^l$, $\mathbf{rem}(u_1) \geq \mathbf{rem}(u_5)$. (b) Since $u_6^r \geq u_2^r$, $\mathbf{rem}(u_6) \geq \mathbf{rem}(u_2)$.

- **ins.** This operation is the dual operator of **rem**. It inserts a node from \mathcal{T}_2 to \mathcal{T}_1 (or equivalently deletes the corresponding node from \mathcal{T}_2). Hence, the

cost function given below is the same with `rem` except that the length is normalized with respect to $l_{max}(\mathcal{T}_2)$.

$$\mathbf{ins}(v) = \left(\frac{v^l}{l_{max}(\mathcal{T}_2)} \right) (1 - v^r) \quad (4)$$

- `ch`. This operation computes the closeness of two nodes u and v based on the differences between their attributes. The cost function resembles the one in [32]. However, an additional constraint forces that the matched branch types are identical. If they differ, the cost is set to 1.

$$\mathbf{ch}(u, v) = \beta_1 \frac{|u^l - v^l|}{\max(u^l, v^l)} + \beta_2 \frac{|u^r - v^r|}{\max(u^r, v^r)} + \beta_3 \frac{|u^\theta - v^\theta|}{\max(u^\theta, v^\theta)} \quad (5)$$

Note that $|u^\theta - v^\theta|$ indicates the acute angle as a consequence of retaining the counter-clockwise ordering of the branches in all alternative representations. In the experiments, we give more weight to the similarity of lengths by setting $\beta_1 = 0.5$ and $\beta_2 = \beta_3 = 0.25$. That is, the contribution of the length to the overall attribute change cost is doubled. There is no excuse other than there is a single length variable whereas there are two location variables.

The time complexity for matching two depth-1 trees using Zhang and Shasha [51] is $O(mn)$, where m and n denote the number of leaves in respective trees. A critical issue in tree edit based shape comparison is how the cost of each edit operation is computed. This cost may dominate over the cost of tree matching as in [38]. In this respect, our approach has two advantages. First, the edit cost computations are negligible. Second, the number of leaf nodes are significantly small as a consequence of our excessively smooth representation.

Matching results for sample shapes are shown in Fig. 8. Correct correspondences are found despite significant articulations (Fig. 8(a) and (b)) and miss-

ing sections (Fig. 8(c) and (d)). Each matching induces a cost of converting one shape tree into another. This cost determines the symmetric distance (dissimilarity) between two shapes.

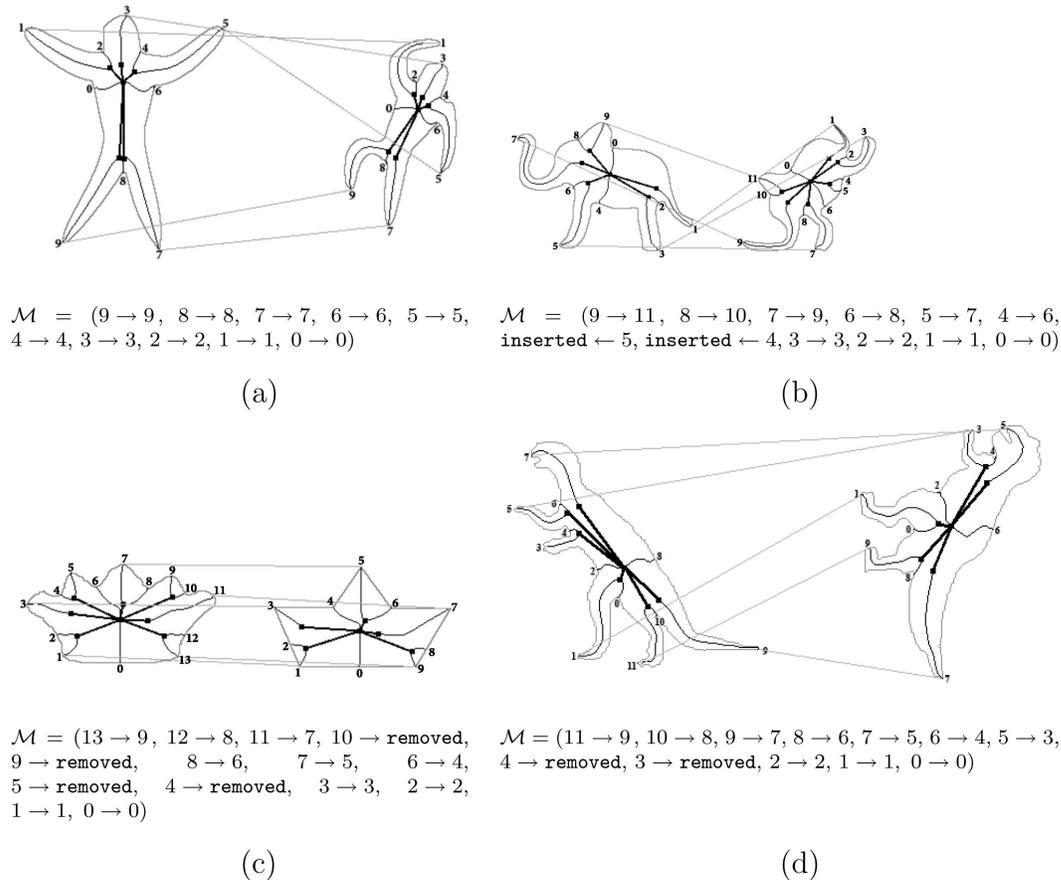


Fig. 8. Some skeletal matching results. Matching costs are 0.683, 1.459, 2.725 and 1.550, respectively. For illustration purposes, matchings are shown only for positive branches.

4 Shape Retrieval Based on Tree Edit Distance

After computing the dissimilarities between shape pairs (total of 180×180), each one of the 180 shapes is selected as a query shape and top twelve shapes in the increasing order of dissimilarity are retrieved. Due to space limitations,

some query results are displayed in Fig. 9 and 10. Number of wrong shapes in the top 5 retrievals are 1, 6, 13, 21, 41 respectively. Average precision-recall curve is given in Fig. 11(a). These results are comparable to the results reported in [2]. The precision is around 87% even when the recall is 100%. Generally speaking, skeletons parsed into depth-1 trees is a good candidate for describing shapes.

The horse and the sea turtle categories have the worst retrieval performance. The average precision-recall curves for them are presented in Fig. 11(b). Typically, mismatches can be organized in four groups discussed below.

4.1 Structurally Different Shapes with Similar Shape Trees

In Fig. 12, one can observe the resemblance between the trees of an elephant and a kangaroo when branches 4 and 5 are deleted from the shape tree of the former. Branches 4 and 5 of the elephant are short. Hence, their deletion costs are small. This small cost gives rise to a shape dissimilarity measure (1.657) which falls in the range of the dissimilarities within the elephant category. The top 8 retrievals for 6 elephant shapes are displayed in Fig. 13. It is very difficult to distinguish the cost of editing an elephant to a kangaroo from the cost of editing an elephant to another elephant.

4.2 Structurally Similar Shapes

Structurally similar shapes have similar shape trees. Notice that the cat and the horse shapes in Fig. 14 have the same number of branches with simi-

	0.487	0.683	0.711	0.752	0.806	1.857	1.931	1.998	2.022	2.030	2.035	
	0.864	1.109	1.378	1.402	1.510	1.658	1.741	1.758	1.806	1.914	1.971	
	0.901	0.906	1.033	1.122	1.657	1.663	1.842	1.848	1.848	1.909	1.946	
	0.542	1.370	1.483	1.522	1.580	1.599	1.671	1.749	1.784	2.232	2.550	
	0.628	1.649	1.655	1.770	1.931	2.275	2.325	2.394	2.427	2.448	2.484	
	1.202	1.205	1.286	1.337	1.591	2.559	2.606	2.750	2.932	3.057	3.090	
	0.897	0.967	1.359	1.455	1.522	1.590	1.815	1.848	1.941	2.028	2.572	
	0.166	0.262	0.692	0.710	0.717	1.477	1.479	1.496	1.517	1.518	1.567	
	1.031	1.033	1.175	1.256	1.518	1.555	1.827	1.908	1.930	1.949	1.958	
	0.374	1.146	1.220	1.372	1.536	1.680	1.745	1.806	1.817	1.821	1.842	
	0.264	1.095	1.098	1.568	1.661	2.551	2.795	2.930	2.964	3.000	3.014	
	0.558	0.783	1.029	1.048	1.049	1.643	1.763	1.814	1.835	2.056	2.066	
	1.261	1.301	1.337	1.596	1.733	2.115	2.163	2.178	2.189	2.197	2.219	
	0.327	0.572	0.985	1.046	1.345	1.869	1.898	1.941	1.952	1.969	2.116	
	1.136	1.156	1.171	1.383	1.395	1.461	1.466	1.502	1.507	1.517	1.521	

Fig. 9. Some query results.

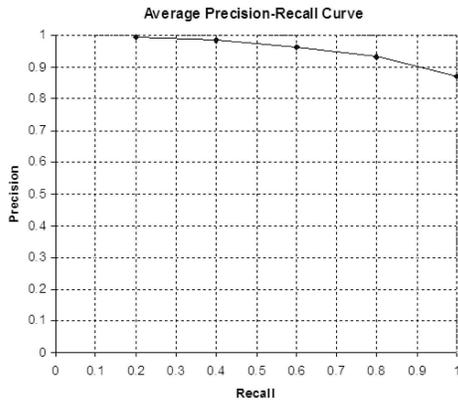
lar characteristics. Therefore, the dissimilarity between a cat and a horse is comparable to the dissimilarities among cats (Fig. 14).

	0.354	0.702	1.071	1.252	1.339	1.461	1.534	1.671	1.692	1.789	1.794
	0.411	0.541	0.547	0.573	0.637	1.959	2.006	2.011	2.052	2.056	2.057
	1.286	1.532	1.536	1.713	1.772	1.825	1.836	1.836	1.843	1.897	1.955
	0.441	0.667	0.681	0.873	0.873	1.492	1.646	1.680	1.691	1.798	1.842
	1.681	1.774	1.951	2.042	2.070	2.503	2.531	2.542	2.597	2.612	2.617
	0.535	0.653	0.747	1.084	1.627	1.672	1.803	1.811	1.831	1.992	2.197
	0.258	0.343	0.520	0.990	1.889	2.238	2.239	2.275	2.361	2.429	2.480
	1.498	2.137	2.293	2.372	2.513	2.559	2.692	2.743	2.747	2.858	2.902
	0.788	1.535	1.781	1.827	1.872	1.906	2.002	2.104	2.171	2.239	2.241
	0.312	0.403	0.593	0.806	0.831	1.816	1.889	1.952	2.057	2.059	2.170
	1.556	1.714	2.290	2.388	2.649	2.658	2.668	2.672	2.689	2.717	2.725
	0.227	0.621	1.670	2.161	2.169	2.198	2.204	2.299	2.414	2.440	2.451
	1.292	1.293	1.363	1.371	1.397	1.409	1.466	1.477	1.488	1.491	1.534
	1.740	1.824	1.831	2.058	2.086	2.243	2.256	2.266	2.287	2.316	2.327
	0.286	1.106	1.189	1.740	1.745	1.814	1.844	1.864	1.901	1.945	1.949

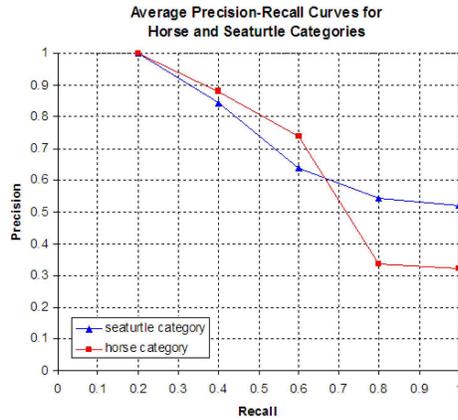
Fig. 10. Some query results.

4.3 Shapes with Spurious Branches

Spurious branches may be observed due to within category variability or they may be an artifact of extreme articulations. These erroneous branches increase

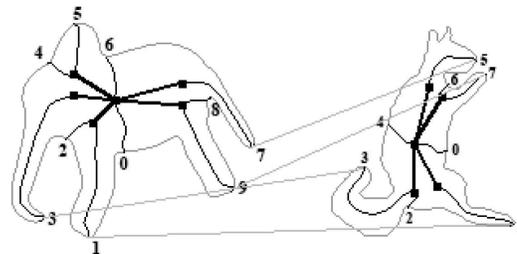


(a)



(b)

Fig. 11. Average precision-recall graphs (a) All shapes (b) The horse and the sea turtle categories.



$$\mathcal{M} = (9 \rightarrow 7, 8 \rightarrow 6, 7 \rightarrow 5, 6 \rightarrow 4, 5 \rightarrow \text{removed}, 4 \rightarrow \text{removed}, 3 \rightarrow 3, 2 \rightarrow 2, 1 \rightarrow 1, 0 \rightarrow 0)$$

Fig. 12. Structurally different shapes with similar shape trees.

the dissimilarity due to extra removal costs. An example is given in Fig. 15.

4.4 Shapes with a Significantly Dominant Branch

In removal and insertion costs, the length of each branch is normalized with respect to the maximum branch length. If there is a branch which is significantly longer than the remaining ones, all the other branches lose their importance. In a key shape shown in Fig. 16, with one exception, all of the branches have comparable lengths. Consequently, relative to the outlier branch, all the re-

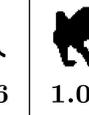
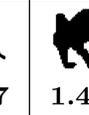
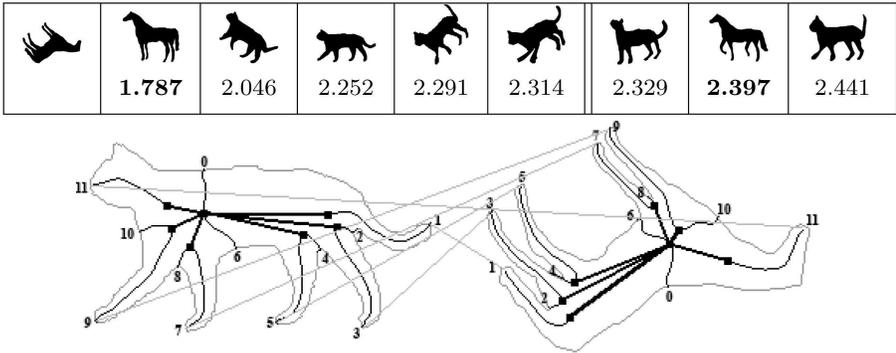
	 0.906	 0.994	 1.040	 1.157	 1.459	 1.492	 1.536	 1.632
	 0.846	 0.994	 1.033	 1.420	 1.580	 1.676	 1.691	 1.695
	 0.901	 0.906	 1.033	 1.122	 1.657	 1.663	 1.842	 1.848
	 1.122	 1.157	 1.420	 1.536	 1.562	 1.772	 1.782	 1.798
	 1.459	 1.562	 1.663	 1.708	 1.744	 1.958	 1.963	 1.967
	 0.846	 0.901	 1.040	 1.536	 1.646	 1.708	 1.733	 1.743

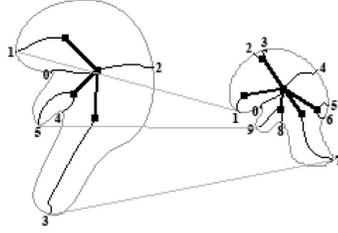
Fig. 13. The top 8 retrievals for 6 elephants. The dissimilarity between an elephant and a kangaroo is comparable to the dissimilarities among elephants.



$$\mathcal{M} = (11 \rightarrow 11, 10 \rightarrow 10, 9 \rightarrow 9, 8 \rightarrow 8, 7 \rightarrow 7, 6 \rightarrow 6, 5 \rightarrow 5, 4 \rightarrow 4, 3 \rightarrow 3, 2 \rightarrow 2, 1 \rightarrow 1, 0 \rightarrow 0)$$

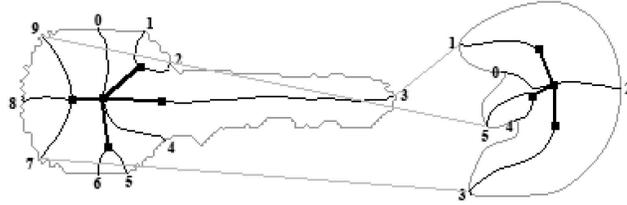
Fig. 14. Structurally similar shapes.

maining branches become insignificant. When this shape is compared to the shape shown in the right, the dissimilarity (1.824) is much lower than our expectation.



$$\mathcal{M} = (5 \rightarrow 9, 4 \rightarrow 8, 3 \rightarrow 7, \textit{inserted} \leftarrow 6, \textit{inserted} \leftarrow 5, 2 \rightarrow 4, \textit{inserted} \leftarrow 3, \textit{inserted} \leftarrow 2, 1 \rightarrow 1, 0 \rightarrow 0)$$

Fig. 15. Computing edit-distance between two artificially generated shapes of the same category. The shape dissimilarity is found to be 1.337 due to erroneous branches.



$$\mathcal{M} = (9 \rightarrow 5, 8 \rightarrow 4, 7 \rightarrow 3, 6 \rightarrow \textit{deleted}, 5 \rightarrow \textit{deleted}, 4 \rightarrow 2, 3 \rightarrow 1, 2 \rightarrow \textit{deleted}, 1 \rightarrow \textit{deleted}, 0 \rightarrow 0)$$

Fig. 16. Shapes with a significantly dominant branch. The dissimilarity between two shapes (1.824) is much lower than our expectation.

5 Matching with Tree Edit in a Context

Assigning a fixed significance to shape forming primitives and their attributes is the primary source of misleading edit distance computations introduced in Section 3. In this section, we modify the importance of primitives by considering them in a context defined for each query. The context is the collection of shapes that are in the category of the database object to be compared with a query shape. The rest of the section is divided into two subsections. First, we present category tree construction. Second, we develop an algorithm for category influenced matching.

5.1 Construction of Category Tree

A category tree is a union of shape trees. Each of its nodes holds a list of attributes collected from the members included in the context of a certain query. Two types of constructions are possible. We refer to them as static and dynamic, respectively.

In the static formation, the shapes to be united are given in advance. For simplicity, the shape tree with the maximum number of nodes is designated as a *base tree*. All the remaining trees are matched to the base tree and the category tree is formed based on these correspondences between a given tree and the base tree. This procedure has two major drawbacks. First, the structure of the category tree is fixed, consequently, addition of a new shape may require a re-formation from a scratch. Second, the procedure does not guarantee the inclusion of all the available information. An illustration is given in Fig. 17.

Dynamic formation procedure (Fig. 18) resembles *Tree-Unions* which has been proposed by Torsello et. al. [43]. Tree unions have been used in learning shape-classes [45] from shock trees [37]. Merging two shock trees may produce a graph. Therefore, certain checks are introduced in order to force a tree structure for the union. In our case, such a control is not necessary. For details, see the technical reports [4, 15].

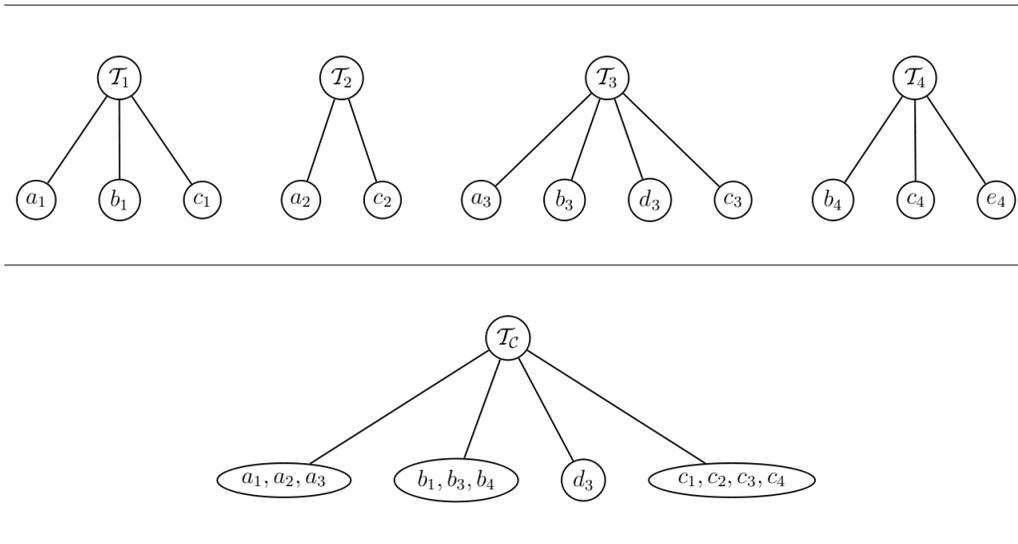


Fig. 17. Static formation of a category tree. \mathcal{T}_3 is the base tree. Matched nodes are labeled with identical letters. Node e_4 in \mathcal{T}_4 is eliminated since it matches with none of the nodes of the base tree.

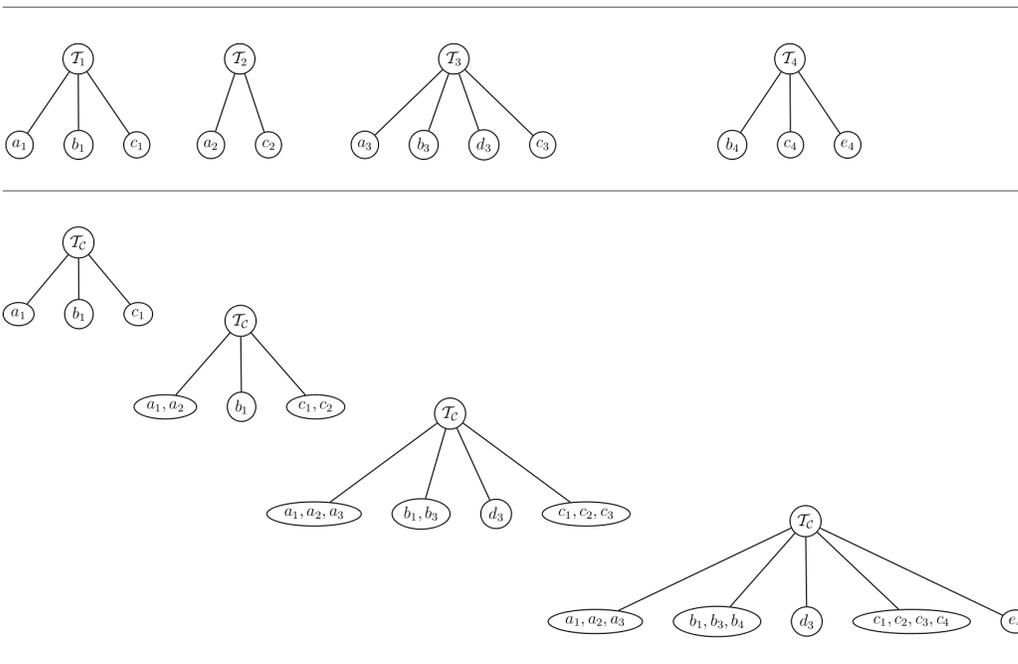


Fig. 18. Dynamic formation of a category tree. The category tree \mathcal{T}_c is enlarged sequentially with the shape trees \mathcal{T}_1 , \mathcal{T}_2 , \mathcal{T}_3 and \mathcal{T}_4 .

5.2 Category Influenced Matching

Let \mathcal{T}_1 denote the tree of a query shape A which is being compared to the tree \mathcal{T}_2 of a database shape B as defined in (1). Each leaf node of a category tree

corresponds to a skeletal branch and holds a list of attributes collected from m category members ($m \leq M$, where M is the total number of shapes in the category). These attributes are:

- the observed ranges for r , θ and l of the branch ($r_{min}, r_{max}, \theta_{min}, \theta_{max}, l_{min}, l_{max}$)
- the frequency of the branch ($freq(\mathcal{B}_k) = m/M$).

Thus, a leaf node \mathcal{B}_k has the following structure

$$\mathcal{B}_k = (\mathcal{B}_k^{r_{min}}, \mathcal{B}_k^{r_{max}}, \mathcal{B}_k^{\theta_{min}}, \mathcal{B}_k^{\theta_{max}}, \mathcal{B}_k^{l_{min}}, \mathcal{B}_k^{l_{max}}, \mathcal{B}_k^{type}, freq(\mathcal{B}_k)) \quad (6)$$

In order to calculate the costs, we define a generic function $f(x|y, [min, max])$ (Fig. 19). In the experiments, we take $\phi_1 = \frac{\pi}{4}$ and $\phi_2 = \frac{4\pi}{9}$. x is defined on the horizontal axis. The function is fixed for a given y value and $[min, max]$. Notice that f value depends not only on the difference $x - y$ but also on the range $[min, max]$. On one hand, when x is within the range, $x - y$ difference is taken as it is. On the other hand, when x falls out of the range, $x - y$ difference is boosted. That is, numerically equal differences are perceived smaller within categories and larger between categories, giving a distance weighted in a context. The idea is not so different than the Mahalanobis distance or the distance defined in [36].

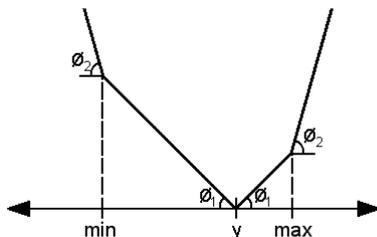


Fig. 19. The generic cost function $f(x|y, [min, max])$.

The modified cost function for $\text{ch}(u, v, \mathcal{B})$ operation is

$$\frac{f(u^r|v^r, \mathcal{B}) + f(u^\theta|v^\theta, \mathcal{B}) + 3f(u^l|v^l, \mathcal{B})}{5} \times \text{freq}(\mathcal{B})$$

Similarly, the cost functions for rem and ins operations are defined as

$$\text{rem}(u) = \left(\frac{u^l}{l_{\max}(\mathcal{T}_1)} \right) (1 - u^r) \quad (7)$$

$$\text{ins}(v) = \left(\frac{v^l}{l_{\max}(\mathcal{T}_2)} \right) (1 - v^r) \times \text{freq}(\mathcal{B}) \quad (8)$$

Since the category of the shape \mathcal{B} is known, the insertion cost of a node is multiplied by a factor of a significance associated with the corresponding skeleton branch.

6 Results and Discussions

We perform our experiments on two different shape sets which were shown in Section 1. In order to utilize the shape sets as best as we can, we perform N (where N is the number of shapes) independent queries by selecting one of the shapes as a query shape and excluding it from the context of the query. Due to space limitations, only sample retrieval results (Fig. 20, 21) and precision-recall graphs (Fig 22, Fig 23) are given.

For the small set, the precision values are above 99.4% for all the recall levels. Notice that there is a significant improvement in the problematic categories (such as the horse and the sea turtle) which were discussed in Section 4. Further notice that the dissimilarities at the 6th retrievals are significantly above the dissimilarities at the 5th retrievals, yielding a very good within group ver-

sus between group separation. The new dissimilarity measure is more intuitive than the measure obtained without considering the context. Compare Fig. 24 with Fig. 13. The elephant shape shown in Fig. 25(b) has two more branches than the other elephant shapes in the shape set. The ordinary dissimilarity between this shape and the elephant shape shown in Fig. 25(a)(the query shape) is 1.459 due to these extra branches. Yet, in the category-influenced matching, less attention is paid to these additional branches since corresponding primitives occur rarely in the elephant category. Therefore, the dissimilarity measure decreases to 0.693. When the elephant shape shown in Fig. 25(b) is selected as the query shape, the cost is higher (1.004), as expected.

When the experiment is conducted on the larger set, the importance of context becomes more visible at high recall levels (Fig 23). At 100% recall, the improvement obtained by context information is more striking in the enlarged shape set (50% improvement in the precision) compared to the improvement in the small shape set. However, at the lowest recall levels, context information may even be harmful. This is due to the existence of categories with quite high within group variability. If we compare Bull’s eye scores for matching results without and with context, we see that in the former case, 243 of the categories lead to 100% Bull’s eye score, whereas this number increases to 499 in the former case. At the same time, the number of categories with less than 6% Bull’s eye score also increases from 7 to 22.

In our second experiment, we exploit structural equivalence between the shape and the category trees in order to facilitate shape to category comparison. We simply replace vector valued attributes of the category tree with the ordinary averages of the components, forming a naive *mean shape tree*. Since a mean shape tree is undistinguishable from a shape tree, we can apply the matching

algorithm given in Section 5.2 in order to determine the dissimilarity between a query shape and a collection of shapes forming a category. We make two remarks. First, the mean tree is uniquely defined as an ordinary average. Hence, it differs from a mean or median structure which has equal edit distances to all the contributing shapes as in [11,20]. Second, a comparison of a shape tree with a mean tree is steered by the category tree from which the mean tree is calculated. Even though the mean tree does not capture within group variability, the category tree does. When shape to category comparison experiment is conducted on the smaller set, for 178 shapes, the correct category is retrieved at the first retrieval. For only one shape (squirrel), the correct category can not be retrieved in the top two or in the top three retrievals. When shape to category comparison experiment is conducted on the larger set, for 824 shapes, the correct category is retrieved at the first retrieval. For 898 shape, the correct category is retrieved at the first or at the second retrieval. For 77 shapes, the correct category can not be retrieved in the top t three retrievals. That is a retrieval rate around 82 – 93% is obtained when our constructions are used for categorization on the larger set. Due to space limitations, the results are given as a Supplementary Material [5].

Asymmetry is an important characteristic of our dissimilarity measure. The asymmetry is also observed in shape to category comparisons (Supplementary Material [5]). The second retrieved category for all of the twenty human shapes is the tulip. However, for non of the tulip shapes, the human is among the top three retrievals. We notice that pairwise similarities within the human category is lower than pairwise similarities within the tulip category. Hence, the human category, having a smaller range and variation, will be less tolerant than the tulip category. Consequently, it will be more difficult to fit a query

shape into this less tolerant category.

Note that we consider a shape only in a group of shapes. The grouping (which is performed externally) is the source of bias. One should be cautious about the fact that dissimilarity calculation is biased by the underlying categorization, yet it is at the same time used in order to assign a category. This is an interesting interplay and may be very valuable in designing adaptive retrieval scenarios where the database is enlarged with the addition of newly categorized shapes. This is an open issue we are currently investigating.

One might be concerned about the fact that the representation is too coarse, thus, it lacks sensitivity. It appears that multiple level representation schemes are necessary to satisfy the opposing goals of sensitivity and stability. It is better to start out with a stable scheme than a sensitive one and add subtle details gradually. Currently, we are exploring a robust scheme for integrating boundary details in a layered computation and performing perturbation analysis on the boundary fragments captured by disconnection points. Rather than adding new cost terms to tree-edit cost function, the influence of secondary terms can be added at higher layers of computation in the line of our previous work [16].

											
	0.239	0.339	0.389	0.414	0.469	2.643	2.657	2.669	3.031	3.095	3.112
											
	0.585	0.784	0.787	0.841	0.941	1.378	1.482	1.560	1.635	1.775	1.823
											
	0.378	0.452	0.457	0.621	0.968	2.456	2.505	2.505	2.571	2.578	2.595
											
	0.459	0.964	0.972	1.070	1.253	2.238	2.325	2.327	2.373	2.379	2.757
											
	0.310	0.658	0.687	0.751	1.108	2.958	2.962	2.967	2.992	3.029	3.048
											
	0.297	0.315	0.447	0.546	0.686	1.829	1.909	1.991	2.021	2.087	2.090
											
	0.482	0.528	0.577	0.606	0.989	2.085	2.098	2.153	2.156	2.167	2.328
											
	0.145	0.185	0.411	0.411	0.426	2.690	2.751	2.841	2.845	2.849	2.872
											
	0.461	0.520	0.672	0.705	0.747	2.464	2.466	2.470	2.719	2.743	2.744
											
	0.114	0.644	0.701	0.814	0.851	1.475	1.508	1.565	1.633	1.652	1.814
											
	0.156	0.422	0.431	0.617	0.747	3.412	3.470	3.519	3.564	3.572	3.593
											
	0.238	0.281	0.333	0.413	0.417	2.508	2.542	2.781	2.805	2.860	2.946
											
	1.063	1.080	1.087	1.213	1.275	1.975	2.015	2.272	2.293	2.339	2.432
											
	0.187	0.255	0.449	0.476	0.552	2.851	2.863	2.866	2.928	2.987	3.029
											
	1.554	1.726	1.747	1.795	1.832	1.870	1.896	1.912	1.935	1.970	1.980

Fig. 20. Some query results for category influenced matching.

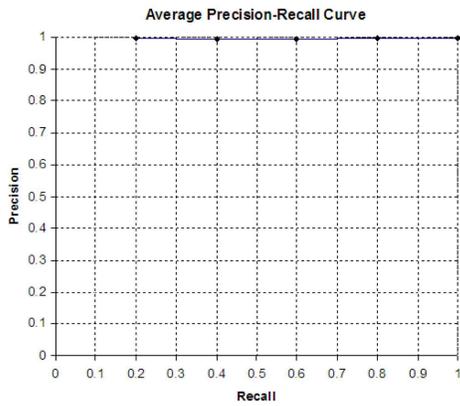
References

- [1] C. Aslan, Disconnected skeleton for shape recognition, Master's thesis, Dept. of Computer Engineering, Middle East Technical University (May 2005).

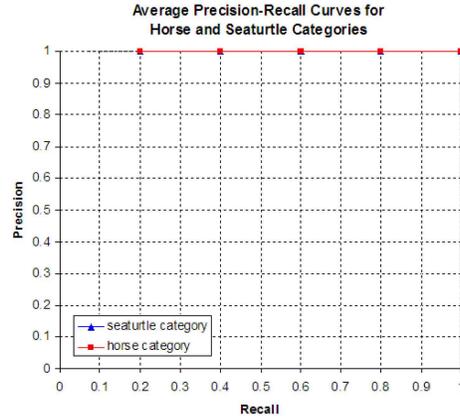
	0.150	0.365	0.460	0.499	0.596	2.277	2.355	2.386	2.399	2.412	2.431
	0.259	0.349	0.365	0.378	0.413	2.245	2.290	2.302	2.397	2.425	2.436
	1.790	1.930	2.063	2.086	2.273	2.588	2.604	2.671	2.676	2.700	2.705
	0.188	0.379	0.466	0.544	0.544	1.855	1.892	1.924	2.038	2.093	2.115
	1.654	1.863	1.907	1.942	2.035	3.026	3.064	3.274	3.428	3.469	3.494
	0.134	0.194	0.202	0.478	0.825	2.314	2.426	2.517	2.865	2.984	2.996
	0.072	0.131	0.175	0.326	0.755	2.685	2.767	2.771	2.791	2.856	2.874
	0.621	0.919	0.949	1.016	1.227	1.856	1.870	1.916	2.075	2.102	2.405
	0.286	0.509	0.887	1.202	1.753	2.649	2.676	2.680	2.795	2.801	2.930
	0.148	0.159	0.276	0.395	0.443	2.284	2.376	2.408	2.538	2.555	2.661
	0.843	0.867	1.281	1.287	1.599	2.838	2.884	2.904	2.940	2.971	3.061
	0.063	0.218	1.060	1.333	1.358	2.745	2.791	2.812	2.843	3.037	3.122
	1.245	1.508	1.544	1.580	1.588	1.608	1.795	1.860	1.940	2.008	2.253
	1.196	1.428	1.533	1.557	1.559	2.595	2.867	2.927	2.941	2.980	3.021
	0.286	0.693	0.699	0.863	0.871	2.522	2.537	2.689	2.767	2.810	2.867

Fig. 21. Some query results for category influenced matching.

- [2] C. Aslan, A. Erdem, E. Erdem, S. Tari, Disconnected skeleton: Shape at its absolute scale., IEEE Trans. Pattern Anal. Mach. Intell. (to appear).



(a)



(b)

Fig. 22. Average precision-recall graphs (a) All shapes (b) Horse and sea turtle categories.

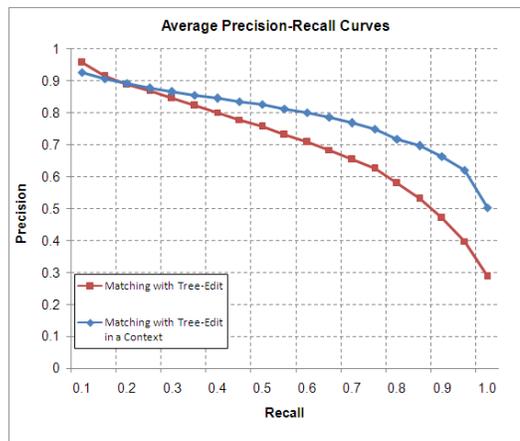


Fig. 23. Average precision-recall graphs for the enlarged set.

- [3] C. Aslan, S. Tari, An axis-based representation for recognition, in: ICCV, 2005.
- [4] E. Baseski, Context-sensitive matching of two shapes, Master's thesis, Dept. of Computer Engineering, Middle East Technical University (July 2006).
- [5] E. Baseski, A. Erdem, S. Tari, Supplemental material.
- [6] R. Basri, Recognition by prototypes, in: CVPR, 1993.
- [7] M. Bicego, V. Murino, M. Figuerido, Similarity-based classification of sequences using hidden Markov models, Pattern Recognition 37 (12) (2004) 2281–2291.

	 0.356	 0.414	 0.418	 0.431	 0.693	 2.132	 2.140	 2.149
	 0.375	 0.446	 0.450	 0.514	 0.815	 2.100	 2.129	 2.129
	 0.378	 0.452	 0.457	 0.621	 0.968	 2.456	 2.505	 2.505
	 0.675	 0.756	 0.828	 0.848	 0.857	 2.224	 2.240	 2.355
	 0.898	 1.004	 1.106	 1.170	 1.254	 2.188	 2.193	 2.265
	 0.401	 0.467	 0.480	 0.615	 0.908	 2.394	 2.422	 2.494

Fig. 24. The top 8 retrievals for 6 elephants. Compare the results to the ones in Fig. 13.

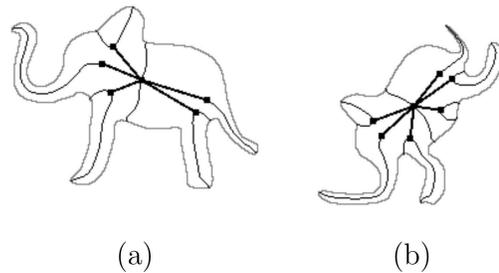


Fig. 25. Shape trees of two elephant shapes having (a) ten leaf nodes (b) twelve leaf nodes.

- [8] M. Bicego, V. Murino, M. Pelillo, A. Torsello, Similarity-based pattern recognition, *Pattern Recognition* 39 (10) (2006) 1813–1814.
- [9] H. Blum, Biological shape and visual science, *Journal of Theoretical Biology* 38 (1973) 205–287.
- [10] H. Bunke, U. Buhler, Applications of approximate string matching to 2D shape

- recognition, *Pattern Recognition* 26 (12) (1993) 1797–1812.
- [11] H. Bunke, X. Jiang, K. Abegglen, A. Kandel, On the weighted mean of a pair of strings, *Pattern Analysis and Applications* 5 (1) (2002) 23–30.
- [12] H. Bunke, K. Shearer, A graph distance metric based on the maximal common subgraph, *Pattern Recognition Letters* 19 (1998) 255–259.
- [13] R. P. W. Duin, D. Ridder, D. M. J. Tax, Experiments with a featureless approach to pattern recognition, *Pattern Recognition Letters* 18 (11) (1997) 1159–1166.
- [14] S. Edelman, F. Cutzu, S. Duvdevani-Bar, Similarity to reference shapes as a basis for shape representation, in: *COGSCI*, 1996.
- [15] A. Erdem, A dynamic procedure for forming shape categories from skeletal shape trees, Tech. rep., METU-CENG-TR-2007-04 (2007).
- [16] A. Erdem, E. Erdem, S. Tari, Articulation prior in an axial representation., in: *International Workshop on the Representation and Use of Prior Knowledge in Vision*, 2006.
- [17] R. Goldstone, D. Medin, J. Halberstadt, Similarity in context, *Memory and Cognition* 25 (1997) 237–255.
- [18] F. Grandidier, R. Sabourin, C. Y. Suen, Integration of contextual information in handwriting recognition systems, in: *ICDAR*, 2003.
- [19] B. Günsel, A. M. Tekalp, Shape similarity matching for query-by-example, *Pattern Recognition* 31 (7) (1998) 931–944.
- [20] X. Jiang, A. Muenger, H. Bunke, Computing the generalized mean of a set of graphs, in: *Workshop on Graph Based Representations*, 1999.

- [21] L. J. Latecki, R. Lakamper, Shape similarity measure based on correspondence of visual parts, *IEEE Trans. Pattern Anal. Mach. Intell.* 22 (10) (2000) 1185–1190.
- [22] M. Leyton, A process-grammar for shape, *Artificial Intelligence* 34 (2) (1988) 213–247.
- [23] T. Liu, D. Geiger, Approximate tree matching and shape similarity, in: *ICCV*, 1999.
- [24] P. Lombardi, V. Cantoni, B. Zavidovique, Context in robotic vision: Control for real-time adaptation, in: *ICINCO*, 2004.
- [25] D. Mumford, Mathematical theories of shape: Do they model perception?, in: *SPIE*, vol. 1570, 1991.
- [26] M. Neuhaus, H. Bunke, Edit distance-based kernel functions for structural pattern classification, *Pattern Recognition* 39 (10) (2006) 1852–1863.
- [27] M. Neuhaus, H. Bunke, Automatic learning of cost functions for graph edit distance, *Inf. Sci.* 177 (1) (2007) 239–247.
- [28] H. Nishida, A structural model of shape deformation, *Pattern Recognition* 28 (1995) 1611–1620.
- [29] H. Nishida, Robust structural indexing through quasi-invariant shape signatures and feature generation, in: *IWVF*, 2001.
- [30] I. Omer, M. Werman, Image specific feature similarities., in: *ECCV*, 2006.
- [31] E. Pekalska, R. P. W. Duin, *The Dissimilarity Representation for Pattern Recognition. Foundations and Applications*, World Scientific, 2005.
- [32] M. Pelillo, K. Siddiqi, S. W. Zucker, Matching hierarchical structures using association graphs, *IEEE Trans. Pattern Anal. Mach. Intell.* 21 (11) (1999) 1105–1120.

- [33] F. Pernus, A. Leonardis, S. Kovacic, Two-dimensional object recognition using multiresolution non-information-preserving shape features, *Pattern Recognition Letters* 15 (11) (1994) 1071–1079.
- [34] C. D. Ruberto, Recognition of shapes by attributed skeletal graphs, *Pattern Recognition* 37 (2004) 21–31.
- [35] P. G. Schyns, R. L. Goldstone, J. P. Thibaut, The development of features in object concepts, *Behavioral and Brain Sciences* 21 (1998) 1–54.
- [36] S. Sclaroff, Deformable prototypes for encoding shape categories in image databases, *Pattern Recognition* 30 (4) (1997) 627–642.
- [37] T. B. Sebastian, P. N. Klein, B. B. Kimia, Shock-based indexing into large shape databases, in: *ECCV*, 2002.
- [38] T. B. Sebastian, P. N. Klein, B. B. Kimia, Recognition of shapes by editing their shock graphs, *IEEE Trans. Pattern Anal. Mach. Intell.* 26 (5) (2004) 550–571.
- [39] K. Siddiqi, A. Shokoufandeh, S. J. Dickinson, S. W. Zucker, Shock graphs and shape matching, *Int. J. Comput. Vision* 35 (1) (1999) 13–32.
- [40] M. Sonka, V. Hlavac, R. Boyle, *Image Processing Analysis and Machine Vision.*, Chapman and Hall, 1993.
- [41] B. J. Super, Knowledge-based part correspondence, *Pattern Recognition* 40 (10) (1997) 2818–2825.
- [42] S. Tari, J. Shah, H. Pien, Extraction of shape skeletons from grayscale images, *CVIU* 66 (2) (1997) 133–146.
- [43] A. Torsello, E. R. Hancock, Matching and embedding through edit-union of trees, in: *ECCV*, 2002.
- [44] A. Torsello, E. R. Hancock, A skeletal measure of 2d shape similarity, *CVIU* 95 (1) (2004) 1–29.

- [45] A. Torsello, E. R. Hancock, Learning shape-classes using a mixture of tree-unions, *IEEE Trans. Pattern Anal. Mach. Intell.* 28 (6) (2006) 954–967.
- [46] R. Veltkamp, M. Hagedoorn, State-of-the-art in shape matching, Tech. Rep. UU-CS-1999-27, Utrecht University the Netherlands (1999).
- [47] P. Viola, M. Jones, Rapid object detection using a boosted cascade of simple features, in: *CVPR*, 2001.
- [48] R. Wagner, M. Fischer, The string-to-string correction problem, *Journal of the ACM* 21 (1) (1974) 168–173.
- [49] S. Watanabe, *Pattern Recognition: Human and Mechanical*, John Wiley and Sons, 1969.
- [50] L. Wolf, S. Bileschi, A critical view of context, *Int. J. Comput. Vision* 69 (2) (2006) 251–261.
- [51] K. Zhang, D. Shasha, Simple fast algorithms for the editing distance between trees and related problems, *SIAM J. Comput.* 18 (6) (1989) 1245–1262.

Author Biographies

Emre Baseski. Emre Baseski received B.S. and M.S. degrees from Middle East Technical University, Computer Engineering Department in 2003 and 2006 respectively. Currently, he is pursuing his Ph.D. degree at University of Southern Denmark. His main research interest is 3D visual object learning.

Aykut Erdem. Aykut Erdem received B.S. and M.S. degrees from Middle East Technical University, Computer Engineering Department in 2001 and 2003 respectively. Currently, he is pursuing his Ph.D. degree at the same institution. His main research interest is categorization of 2D shapes and shape retrieval in very large databases.

Sibel Tari. Sibel Tari received her B.S. degree from Hacettepe University Computer Science Engineering in 1989 and Ph.D. degree from Northeastern University in 1997. Currently, she is an Assoc. Prof. at the Department of Computer Engineering at Middle East Technical University. Her research interest is in symmetry based shape representation, variational and PDE methods in image and shape analysis.

* **Supplementary Material**

[Click here to download Supplementary Material: sup_material.pdf](#)

Supplementary Material

[Click here to download Supplementary Material: SUP-categorization_results.pdf](#)