Shape Matching By Part Alignment Using Extended Chordal Axis Transform

Zahraa Yasseen, Anne Verroust-Blondet, Ahmad Nasri

Shape Matching By Part Alignment Using Extended Chordal Axis Transform

Z. Yasseen[a,*], A. Verroust-Blondet[a], A. Nasri[b]

[a]*Inria Paris 2 rue Simone Iff CS 42112 75589 Paris Cedex 12 France*
[b]*College of Computing, Fahad Bin Sultan University, KSA.*

Abstract

One of the main challenges in shape matching is overcoming intra–class variation where objects that are conceptually similar have significant geometric dissimilarity. The key to a solution around this problem is incorporating the structure of the object in the shape descriptor which can be described by a connectivity graph customarily extracted from its skeleton. In a slightly different perspective, the structure may also be viewed as the arrangement of protruding parts along its boundary. This arrangement does not only convey the protruding part's ordering along the anti clockwise direction, but also relates these parts on different levels of detail. In this paper, we propose a shape matching method that estimates the distance between two objects by conducting a part-to-part matching analysis between their visual protruding parts. We start by a skeleton-based segmentation of the shape inspired by the Chordal Axis Transform. Then, we extract the segments that represent the protruding parts in its silhouette on varied levels of detail. Each one of these parts is described by a feature vector. A shape is thus described by the feature vectors of its parts in addition to their angular and linear proximities to each other. Using dynamic programming, our algorithm finds a minimal cost correspondence between parts. Our experimental evaluations validate the proposition that part correspondence allows conceptual matching of precisely dissimilar shapes.

Keywords: Shape descriptors, shape matching, salient features, ordered part correspondence, Chordal Axis Transform, dynamic 2D shape segmentation, Dynamic Programming.

1. Introduction

Shape descriptors and similarity measures have been a central research issue for decades. One of the main motivations is to obtain a retrieval mechanism over a database of images. Proposed approaches fall into numerous categories [46] such as contour–based, region–based, transforms, moments, scale space, or spatial interrelation. Many challenges and requirements withstand shape descriptors. One with vital importance and considerable difficulty is tolerance for intra–class variety. Objects may be labeled the same whilst having disparate geometries (see Fig. 1). While the tails of the two airplanes in Fig. 1 (or the backs of the two chairs) have dissimilar shapes, they match within their contexts. Their geometric properties such as size and angle of protrusion become relative to the other protruding parts of the shape. This part alignment process allows matching conceptually similar but precisely dissimilar shapes. Although the part correspondence shown in Fig. 1 is trivial for a human, it is a composite process of dynamic segmentation, numeric representation, and search for an optimal solution when it comes to machines. The part that is concerned with dynamic segmentation also belongs to human intuition. In fact, many studies in cognitive science support the theory that humans visually segment objects into simpler parts [10, 27]. Propositions concerning the visual segmentation process draw relations with geometric primitives [8], negative minima of curvature along the boundary [7, 44],

---

*Corresponding Author: Zahraa Yasseen; Email, zyasseen@gmail.com

or skeleton–based segmentation [35] of the shape [1]. Moreover, these studies grant theories concerning numeric representations that contribute in the segmentation process. Cohen and Singh [10] conducted a series of psychophysical experiments to quantify the influences of segment length, turning angles at part boundaries, and part protrusion on segment identification. De Winter and Wagemans [44] analyzed the role of curvature singularities, symmetry, elongation, proximity, good continuation, and cognitive influences on rules of part formation. Bertamini and Wagemans [7] presented a review on evidence from behavioral, neurophysiological, imaging, and developmental studies to conclude that "convexity and concavity information along a contour is something that the visual system computes and represents as part of how shape is analyzed".
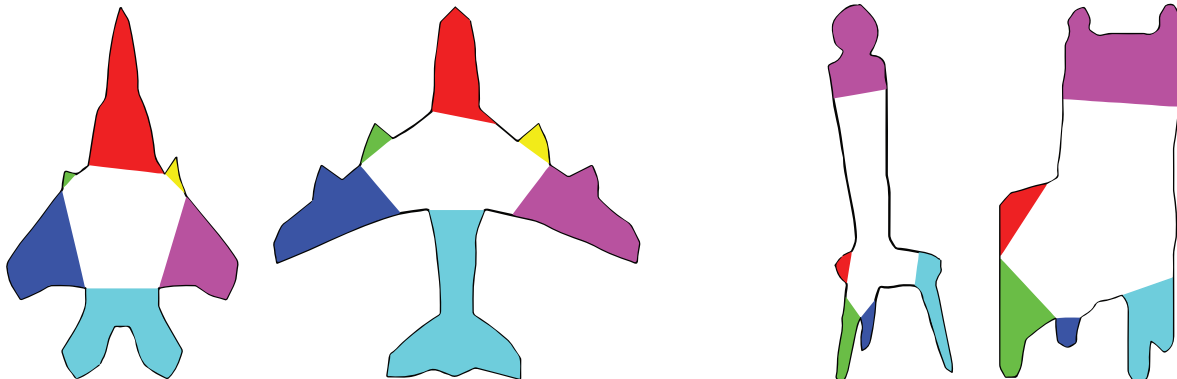

Figure 1: Matching shapes by aligning their visual parts allows corresponding parts that are conceptually similar but precisely dissimilar.

Human perception is known to easily overcome imprecision and occlusion [30]. Hence, we presume a role for these theories while attempting to match the silhouettes of objects taken from different perspectives, with part occlusion, or intra–class diversity. This role helps analyze shapes from the same perspective in which they are perceived by a human inspector: part–based representation independent of level of detail. In this paper, we propose a shape matching scheme inspired by the aforementioned theories and validate its correctness on benchmark datasets. We compute both a skeletal description and a dynamic 2D shape decomposition.

The input data is a binary image representing the silhouette of a single object. We extract the contour, locate corner points, and sample the in–between contour fragments uniformly. The advantage of locating corner points is the inclusion of the sharp features in the sample set. The region is then triangulated using constrained Delaunay triangulation (CDT). The rectified chordal axis transform [29] (CAT) which is a descendant of the medial axis transform, and a set of pruning and merging operations provide a skeleton with an association between skeletal segments and subregions. Using this skeleton, we segment the shape and embed the segmentation in a hierarchy where the leaves are the protruding parts corresponding to the terminal segments defined by the CAT.

The subtrees in this hierarchy group terminal segments with their adjacent internal ones. After pruning away *larger* subtrees using some heuristics, the remaining ones represent the features of the shape on different levels of detail. We refer to all such subtrees and leaves as the *visual parts* that may

---

[1]Hoffman and Richards [20] introduced the 'minima rule' which states that human vision defines part boundaries at negative minima of curvature on silhouettes. Biederman [8] proposed a recognition–by–components theory that assumes the existence of a finite set (total of 36) of simple geometric components that make up an object-part vocabulary. Cave and Kosslyn [9] questioned whether the segmentation is performed prior to recognition and aggravate the role of spatial relations among components. Neri [27] presented experimental evidence to support the recognition-by-part hypothesis as opposed to recognition-as-a-whole. Delorme et al. [13] studied the influence of the presence of diagnostic animal parts and their relative spatial configuration on visual processing. Higgins and Stringer [19] proposed a role for independent motion of body parts in its segmentation.

contribute to the matching process.

Visual Parts are always arranged in their anti–clockwise order of appearance along the boundary of the object. Their spatial and angular distances comprise an inter–distance matrix relating every pair of them. Each *visual part* is described by a set of geometric attributes including relative size, eccentricity, circularity, and convexity in addition to a shape signature. The geometric attributes are multiplied by weights and assembled into a feature vector. The distance between two visual parts is then defined by the Euclidian distance between feature vectors plus the squared distance between signatures. The matching process is performed by topologically rotating the position of the 'first part' in the shape. Consequently, more acceptable segmentations of an object will result from matching it to those in the same class. For every configuration, we employ dynamic time warping (DTW) to find a minimal cost match.

Basically, the components of our work are a skeleton–based segmentation, geometric salience measures to describe visual parts and their spatial relations, and a matching scheme built on a part–to–part correspondence between shapes. The shape descriptor is invariant under affine transformations such as scaling, translation, and rotation. It has also demonstrated tolerance for occlusion, articulation, and minor deformations. The contributions of our work are as follows:

- We bring the CAT into shape description and strengthen some of its weaknesses that are revealed by dissimilarities between skeletons of objects of the same class. It is worth mentioning that Prasad [29] does not provide a matching technique adapted to the CAT representation method.

- We alter the customary practice followed when using DTW in closed shapes by rotating visual parts instead of sample points to find the best match.

The matching process based on the protruding parts is simple and visually intuitive compared with the ones based on skeleton matching. Moreover, this matching process can compare two shapes at different levels of detail or with different topology in a unified way.

The rest of the paper proceeds as follows. We discuss related work that describe shapes by parts or use their skeletons in the process in Section 2. We present our contribution to the CAT segmentation in section 3, whereas the topological structure that embeds different levels of details is discussed in Section 4. The geometric properties, saliency, and distance measure of parts are detailed in section 5. In section 6, we describe the DTW method, and present the retrieval results obtained on Kimia-99 and Kimia-216 datasets [33]. Section 7 presents a brief conclusion of the work presented in this paper.

2. Related Work

Hoffman and Richards [20] introduced the 'minima rule' which states that human vision defines part boundaries at negative minima of curvature on silhouettes. Following the 'minima rule' [20], most of the fragment–based shape descriptors [12, 23, 25, 36, 43] extract boundary segments by locating critical points based on convexity or curvature variations. Latecki et al. [23] partition the contour of a shape into an ordered set of convex and concave boundary arcs. In a somewhat similar way to our approach, they assume that a single visual part of one curve can correspond to a sequence of consecutive parts of the second curve and employ dynamic programming to find a minimal cost correspondence. Sun and Super [36] extract redundant and overlapping contour segments between all the pairs of critical points at extremal curvature values. They represent the segments in a vector space where, after applying PCA, they note that the first few basis 'approximately capture height, sideways shearing, diagonal stretching, and inflation/deflation of a segment.' Daliri et al. [12] extract critical points from the local extrema of the scale invariant curvature profile and construct a dictionary of 11 symbols representing types of contour fragments. The symbols signify curve parts, straight lines, sharp angles, concave and convex parts and such. Guo et al. [18] introduced the notion of 'part importance' based on reconstruction quality, variation, and uniqueness and measured by some conditional entropy. Ma and Latecki [25] propose a shape descriptor with basic geometric units similar to the shape context [6] and suitable for matching of edge fragments. Applying discrete contour evolution [2], Wang et al. [43]

locate critical points to fragment the shape's contour. They use the bag of words model to represent a shape described by 400 overlapping fragments with a shape context descriptor of 300 bins each. Laiche et al. [22] explore the least squares model to represent a shape by a set of normalized cubic curves.

Skeleton–based shape descriptors either require a tree or graph matching method [14, 31, 32, 34, 37, 39, 40] or escape the incurred instabilities by just drawing on topological characteristics provided by the skeleton. Instead of directly matching skeletons, Xie et al. [45] use the MAT to find an optimal point correspondence between two shapes. Goh [17] presents a discussion on strategies that should be adopted in skeleton–based shape matching. Namely, he stresses on the importance of dynamic part–decomposition, visual saliency measures, and the use of both global and local measures. Aslan et al. [1] represented symmetry branches by vectors from an origin located by an evolution process to the disconnection points where the limbs connect to the body. Supported by perceptual motivation, Temlyakov et al. [38] propose strategies to improve shape classification. They identify thin and elongated strand structures and their base structure and measure similarity between shapes on these structures separately. Feldman et al. [15] propose a skeleton that implies part decomposition. Similarity is estimated by the probability that one shape would 'grow' from the skeleton of the other. Lin et al. [24] propose a hybrid shape descriptor that draws on skeletal and contour properties of the shape.

Bai et al. [4] propose a matching method that views the skeleton as a set of shortest paths between pairs of end points. They apply a skeleton pruning process on the medial axis based on discrete curve evolution [2]. In a subsequent publication [3], the skeleton paths shape descriptor is used in conjunction with contour fragments to design a shape classifier. Using a similar endpoints representation for the shape skeleton, Giang et al. [16] further compare tuples of terminal nodes. They define a hyper–graph that relates a node to its k–nearest neighbors.

It is quite evident that the skeleton embeds crucial information about the shape. The above methods provide a rich set of ideas in this regard. In addition to Goh's elaborate study [17] on important features that should be incorporated in any matching method, it seems that there is a general consensus on the distinguished role of endpoints of the skeleton or the terminal parts of the shape.

3. Skeletonisation and Segmentation

The skeleton that we seek is different from the medial axis transform (MAT) of a shape. This skeleton is not only used to depict the axial representation of the shape, but must also provide an intuitive segmentation. In other words, we seek a skeleton–based segmentation where the main visually protruding parts correspond to regions or to union of regions of this segmentation. Although the MAT does not provide such a part–to–axial segment correspondence, the CAT, which is based on principles borrowed from the MAT construction, provides a skeleton and an associated segmentation conjointly. In this section, we give a brief explanation on how the CAT segments and skeletonizes 2D shapes. We also shed some light on inadequacies of the CAT in its current form [29] and propose various amendments to qualify it for shape description and matching.

3.1. The CAT in Its Present Form

The medial axis transform (MAT) of a 2D shape bounded by its contour line C is the locus of the centers of disks interior to the shape and tangent to C in two or more points. The formation of the chordal axis transform (CAT) is based on the observation that the maximal discs used in MAT identify the triangles of the constrained Delaunay triangulation (CDT) of the shape's interior. In a CDT, all vertices lie on the boundary of the shape, giving rise to three types of triangles: *terminal*, *sleeve*, and *junction* (see Fig 2). A *terminal* triangle has two of its edges on the shape's boundary in addition to an interior edge. *Sleeve* and *junction* triangles have two and three internal edges respectively. The term *chord* is used to refer to internal or non-boundary edges in the CDT.

In its primitive form, the CAT is constructed by joining the midpoints of the chords and the centroids of junction and terminal triangles of the CDT. In addition to the lack of an evident segmentation,
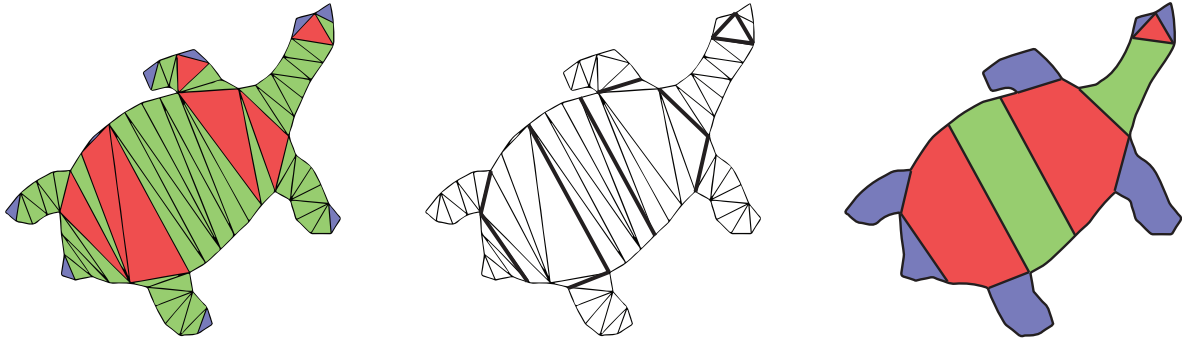
Figure 2: Left to right: the constrained delaunay triangulation (CDT), junction (red), sleeve (green), and terminal (blue) triangles, the chords in the strength profile shown in a darker color, and the segmentation of the shape to three types of segments: junction (red), sleeve (green), and terminal (blue).
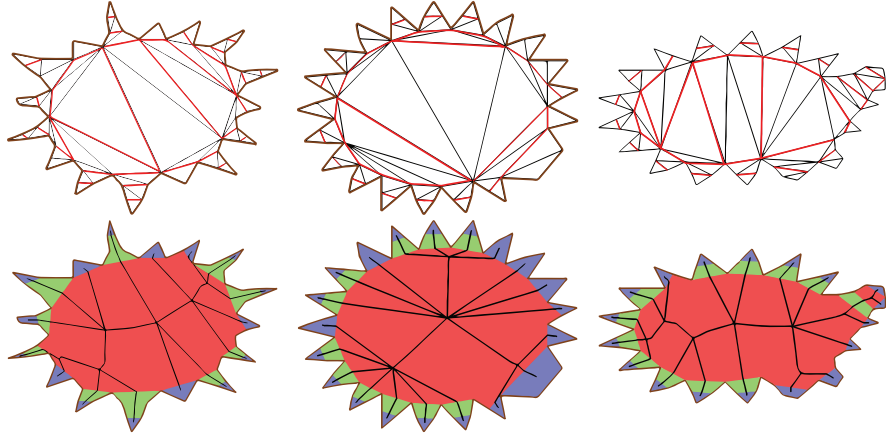
the main problem with this version of the CAT is the exclusive third degree topology of junction points located at the centroids of junction triangles. The *rectified* CAT [29] approaches a solution to this problem by using polygonal elements instead of triangular elements. These polygons are defined by a subset of the chords of the CDT, denoted by the chords strength profile (CSP), and are classified into three types (see Fig 2). *Terminal* and *sleeve polygons* have one and two chords respectively amongst their edges. *Junction polygons* have three or more chords where the number of chords is equal to the degree of the junction point located at the centroid of the polygon. The skeleton is formed by the segments connecting mid points of the chords in addition to the centroids of junction and terminal polygons. In addition to the skeleton, these polygonal elements provide a segmentation of the shape. In what follows we describe how the subset of chords, the CSP, is selected in order to construct the skeleton and a segmentation of the shape. From here on, we will use the term "segment" or "region" to refer to the polygonal elements.

Prasad's [29] idea for selecting the CSP is based on quantifying the notion of the strength of chords. A chord's strength is measured by estimating how far the quadrilateral formed by its flanking triangles is from being cyclic. The chords in the triangulation are ordered in an anti-clock wise traversal of the boundary. A local maximum in this set is a chord whose strength is greater than both of its neighbors and strictly greater than one of them. These local maxima make the CSP. Once the CSP is formed, the segmentation and the skeleton follow trivially (see Figs. 3(a), 4(a), and 5(a)).
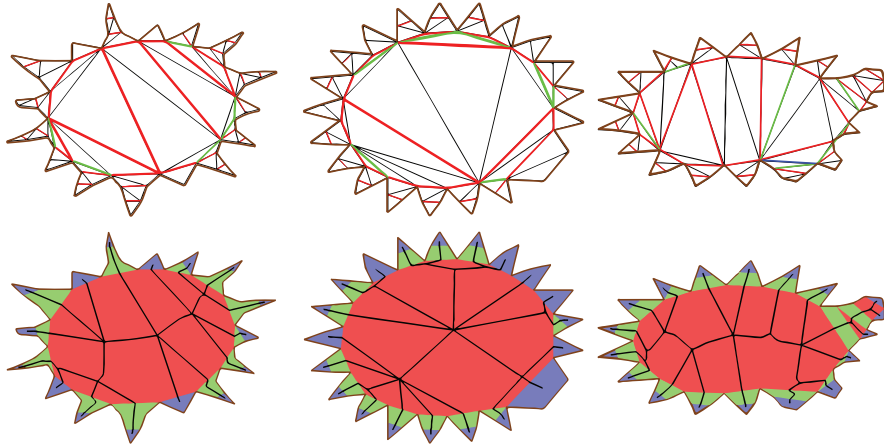
3.2. Extended Chordal Axis Transform

Employing the CAT in shape representation and matching incurs additional requirements of its segmentation and skeletonisation. It also reveals some weaknesses exposed by dissimilar topologies of similar objects. Trivially, it is necessary to have tolerance for an extent of boundary variation where similar objects tend to have similar skeletons. Moreover, some objects are visually segmented into a centralized structure with one junction segment connected to most of the terminal parts (see Fig. 3(a)). Due to its reliance on the CDT, the CAT tends to preclude such junction segments favoring a distributed topology. As we want to keep only significant protruding parts, excessive branching on terminal segments should be avoided and visually unimportant segments should be merged. Nonetheless, in certain instances, terminal segments recede and adjacent junction segments take parts of what visually belong to protruding parts (see tail and fin in Fig. 4(a)).
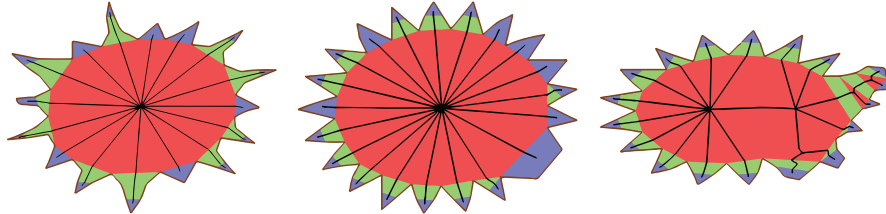
In general, the qualifying properties for a shape descriptor are the invariance to scale, translation, and rotation and tolerance for occlusion, articulation and minor deformations. The CAT is evidently invariant to the affine transformations. However, the second batch of properties is more difficult to accomplish. In our version of the CAT, we tweak its formation phases so as to reduce its sensitivity to minor deformations. Figs. 3, 4, and 5 show multiple examples of objects from the same classes (*star*, *fish*, and *hand*) illustrating the difference between the CAT in [29] and the CAT that we construct in

5

(a) The CSP (the chords of the CDT that are highlighted in red) as proposed in [29] (first row) and the consequent skeleton and segmentation (second row).

(b) The expanded CSP where added chords are the compactness chords shown in green and the enclosing disk chords in blue (first row) and the consequent skeleton and segmentation (second row).

(c) The skeleton and segmentation after applying the pruning, merging, and trimming operations on the segments shown in (b).

Figure 3: The CAT of objects with centralized skeletal topology.

(a) The CSP (the chords of the CDT that are highlighted in red) as proposed in [29] (first row) and the consequent skeleton and segmentation (second row).
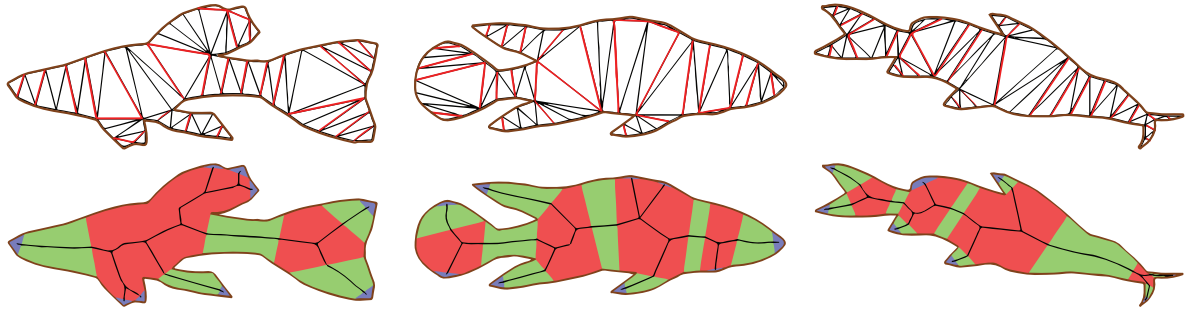

(b) The expanded CSP where added chords are the compactness chords shown in green and the enclosing disk chords in blue (first row) and the consequent skeleton and segmentation (second row).

(c) The skeleton and segmentation after applying the pruning, merging, and trimming operations on the segments shown in (b).

Figure 4: The CAT after expanding the CSP and applying the pruning, merging, and trimming operations results in more restrained junction areas than that proposed in [29]

(a) The CSP (the chords of the CDT that are highlighted in red) as proposed in [29] (first row) and the consequent skeleton and segmentation (second row).

(b) The expanded CSP where added chords are the compactness chords shown in green and the enclosing disk chords in blue (first row) and the consequent skeleton and segmentation (second row).
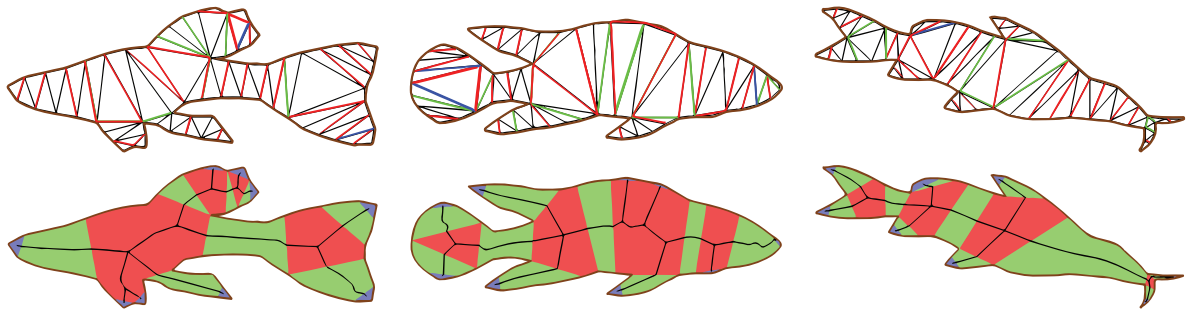

(c) The skeleton and segmentation after applying the pruning, merging, and trimming operations on the segments shown in (b).
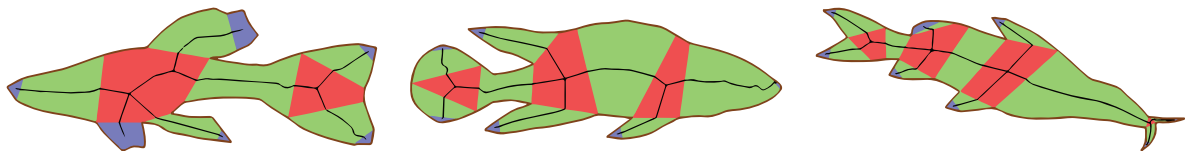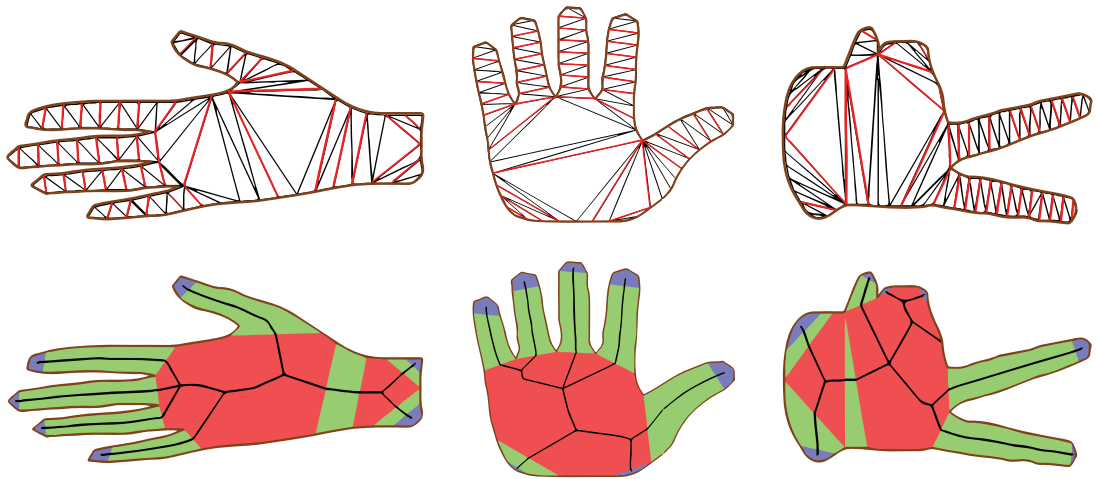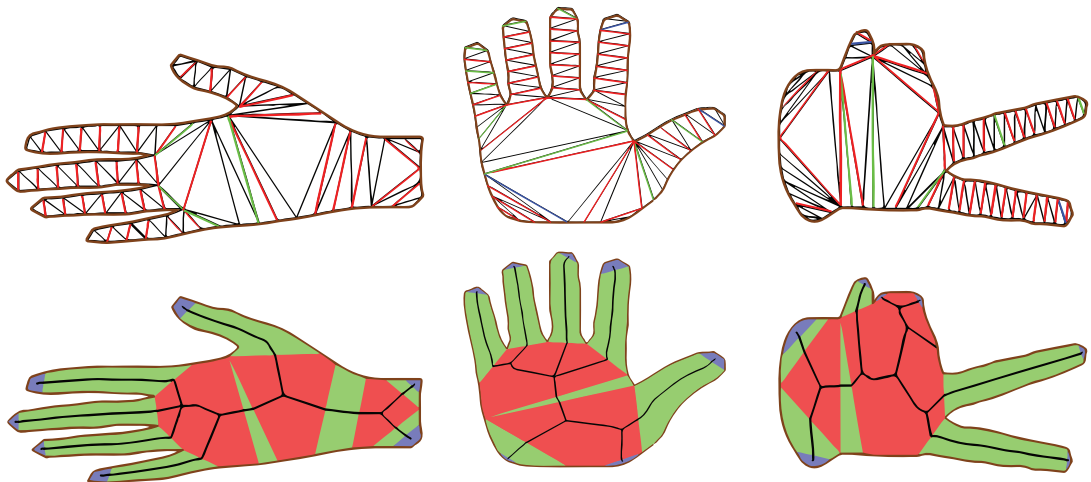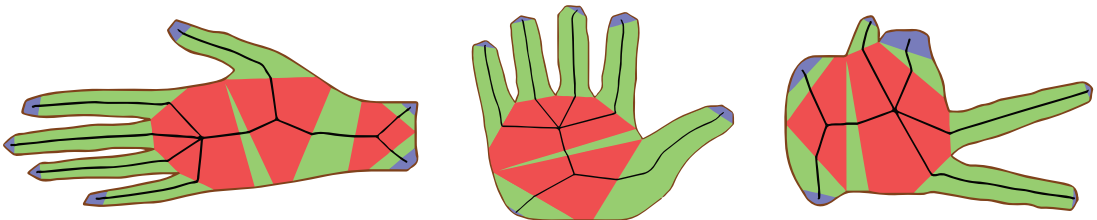
Figure 5: The CAT proposed in this paper results in closer skeletal structures for similar objects than the CAT in [29]. Despite the sever part occlusion in the left most object, its CAT retained most of the structure of the un-occluded object.
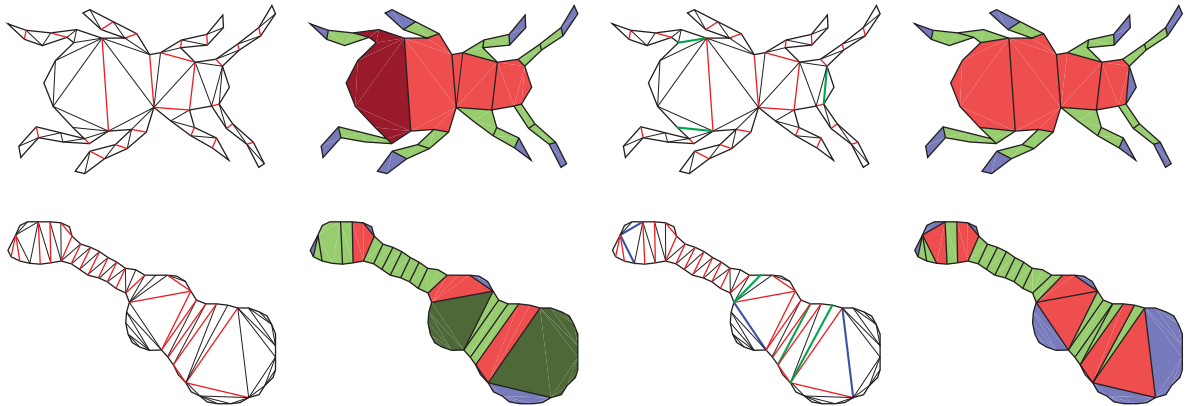
Figure 6: From left to right: chords selected into the CSP in red, the concave region in dark red and the salient parts with round geometry in dark green, the chords with maximal measure of compactness in green and maximal enclosing disk in blue, and the resultant segmentation.

this paper. The figures also show the reduced difference between instances of the modified version of the CAT in the same class of shapes. In particular, the structure of the CAT withstands the sever part occlusion in Fig. 5(c). In the original settings (see Fig. 5(a)) every instance of the CAT relates the four fingers of the hand in a different topology. The previously mentioned shortcomings of CAT can be addressed by interjecting the CAT formation process in two main phases:

1. In the first phase, we expand the CSP to include compactness chords and the enclosing disk chords that will be introduced shortly (see Figs. 3(b), 4(b), and 5(b)).
2. In the second phase, we prune insignificant terminals and analyze blocks of adjacent junction segments for possible merging into fewer higher degree junction segments (see Figs. 3(c), 4(c), and 5(c)).

The two phases added to the CAT formation process allow creating an intermediate finer segmentation that undergoes conditional merging operations. In what follows, we will describe these procedures in detail.

3.2.1. Extended CSP

In certain cases, some visually important chords are not selected into the CSP. This may result in undesirably concave segments (see the dark red segments of the bug in Fig. 6) or neglected salient parts with round geometry (see the dark green segments of the instrument in Fig. 6). The disadvantage of such cases is amplified by the vital role of the visual protruding parts in the shape's description. It is important to associate all visually protruding parts to a segment or a union of adjacent segments. The concave junction segment in the bug and the bulging sleeve of the instrument (Fig. 6) occupy parts of salient visual features. Such cases motivate searching for criteria to segment the shape other than the *strength* of the chord introduced in [29].

The *strength* of a chord is an estimation of how the associated quadrilateral is cyclic. We incorporate the inverse of its circularity ratio denoted as measure of compactness in the CSP selection. In effect, we added more search criteria and expand the CSP to include local maxima of the following values:

1. **Maximal compactness:**
   The measure of compactness of the quadrilateral formed by its flanking triangles as defined in [5]. This is the square of its perimeter divided by its area.
2. **Maximal enclosing disk:**
   The chord splits the boundary of the shape into two parts. Let the chord's boundary segment be the smaller part. The value that we seek here is the number of points on the chord's boundary
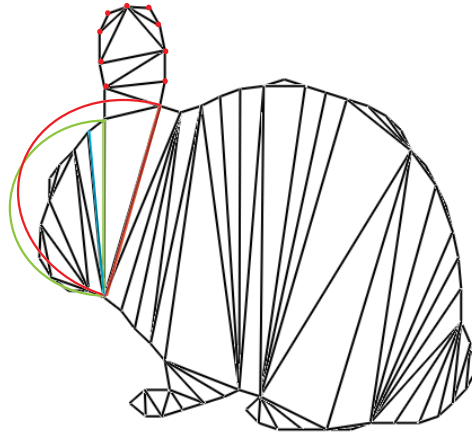
Figure 7: The chord colored in green is a local maximum since its neighbors either have a smaller number of boundary points belonging to their half disks (blue chord) or have some points on their boundary segment outside their half disk (red chord).

segment such that all of them are interior to the disk formed by rotating the chord around its midpoint. Having one point outside this disk sets this value to zero (see Fig. 7).

Having established the CSP, the shape is easily segmented by the chords belonging to this set (see Figs. 3(b), 4(b), and 5(b)).

3.2.2. Insignificant Terminals

Generally speaking, terminal segments represent the parts protruding from the shape with a certain degree of saliency. If a terminal segment, adjacent to a junction segment, does not have sufficiently sharp diversions represented by negatively oriented angles around its boundary, it is tested for insignificance (see Fig. 8). Such a terminal is considered insignificant if the ratio of the length of its chord to the length of its boundary segment exceeds values relative to its size as follows:

1. The area of the terminal segment is smaller than ϵ and the chord length ratio is greater than μ_0.
2. The chord length ratio is greater than μ_1.

where values of ϵ, μ_0, and μ_1 are 1%, 0.75, and 0.9 respectively. Terminal segments with such properties have a flat–like geometry and are thus merged to their neighboring junction segments.

3.2.3. Junction Blocks

Junction segments determine the topology of the skeleton. Whether this topology is centralized or contains symmetrical parts depends on the degrees and connectivity of its junctions. In general, shapes with rotational symmetry have a central junction segment connecting the rather similar visual protruding parts. Nevertheless, the skeleton of a shape with reflectional symmetry is expected to carry the same symmetry as well where mirrored parts have symmetric skeletons. These requirements are not always met by CAT in its original form [29]. For example, the skeleton of the leaf in Fig. 9(a) is deviated from its axis of symmetry by the quasi central sixth degree junction. As such, this junction should be split in a way that allows more symmetry in the skeleton (see Fig. 9(c)). On the other hand, the octopus in Fig. 9(a) has a partial rotational symmetry where approximately similar protruding parts connect to a central unit. Thus, the block of adjacent neighboring junction segments is better merged to one higher degree junction producing a skeleton with a centralized topology (see Fig. 9(c)). The CAT presented in this paper handles these cases by expanding the CSP thus refining the segmentation and then traversing blocks of adjacent junction segments (see Fig. 9(b)) to test possible merge operations. In this section we explain the merging process in detail.
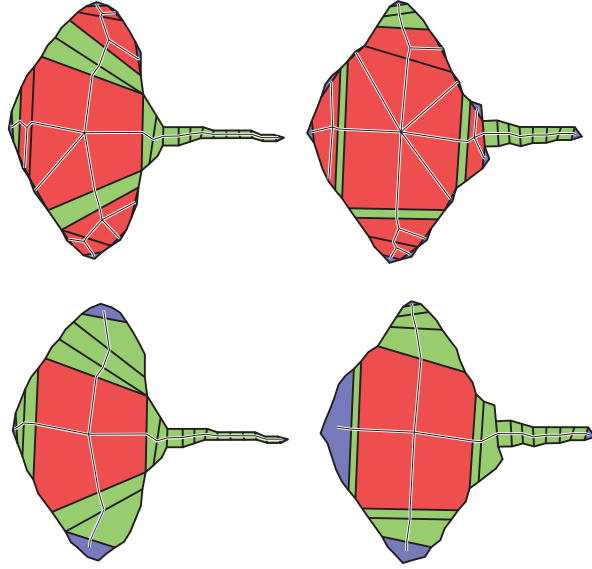
10

Figure 8: The result of merging insignificant terminal segments shown on two objects belonging to the same class. The topological dissimilarity between the two objects decreased significantly after the merging.
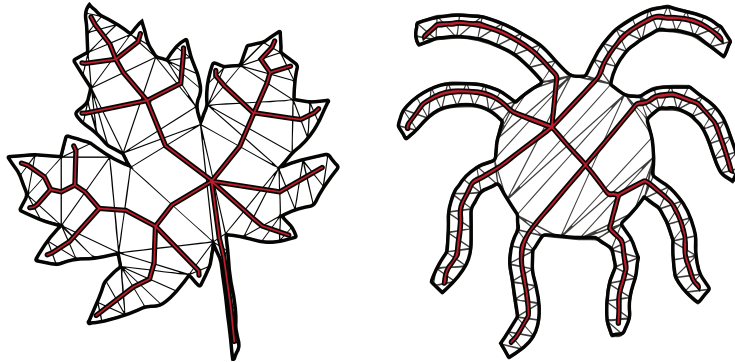
The relation between a chord $\overrightarrow{C}$ and a junction point J can be described by the length of the skeletal segment $\|\overrightarrow{S}\|$ connecting J and the mid point of $\overrightarrow{C}$ in addition to θ, the angle between $\overrightarrow{C}$ and $\overrightarrow{S}$. A small θ or a long $\overrightarrow{S}$ are indications of weakly related chord and junction point. In a block of adjacent junctions, the properties that need to be balanced out for all chords and junction points are the following:

1. Minimum difference between θ and $\pi/2$.
2. Minimal skeletal segment length $\|\overrightarrow{S}\|$. This property prioritizes relating chords to proximal junction points.
3. Minimal junction segments degree.
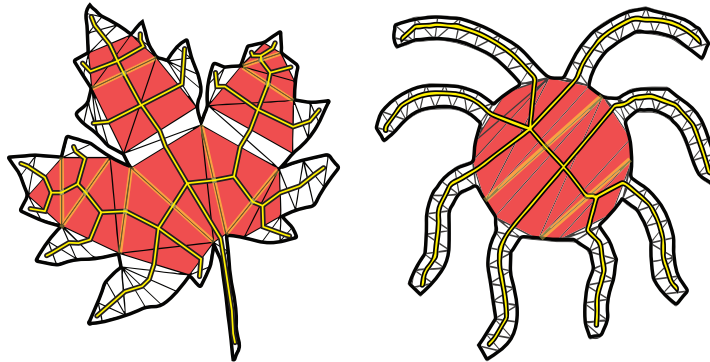4. Minimal chord lengths $\|\overrightarrow{C}\|$.

In a block of adjacent junctions, an optimal setting for the first two properties can always be reached by merging all junctions to one. For example, the largest block of junctions in the leaf in Fig. 9(b) would be merged to one degree 10 junction segment since the collected values of ten skeletal segments and ten angles will definitely be less than that of 24 skeletal segments an 24 angles. To prevent forming this large degree junction segment, we infuse the third condition that is to minimize the degree of produced junction segments. On the other hand, if the block contains a relatively large chord such as the highlighted chords in the octopus of Fig. 9(b), it means that merging its two junctions will produce a junction segment that has less disparate dimensions. For this reason and to balance out the tendency towards smaller junction degrees created by the third condition, we also incorporate the chords' lengths in the fourth condition. To formulate a numerical representation for the above properties, we define the *strain* of a junction segment. Let n be the number of chords of a junction segments and θ_i the angle between the ith chord $\overrightarrow{C_i}$ and the ith skeletal segment $\overrightarrow{S_i}$. The *strain* of this junction is quantified by the following formula:

$$n \times \sum_{i=1}^{n} (\theta_i - \frac{\pi}{2})^2 \times \|\overrightarrow{C_i}\| \times \|\overrightarrow{S_i}\|$$

Since these blocks are normally sets with small cardinalities, it is easy to test all possible combinations of merging. The set that we consider for every attached block of junctions is that of their

11

(a) The CAT as proposed in [29] may fail to depict important shape properties such symmetry and centr.

(b) The segmentation resulting form the extended CSP produces junction segments with smaller degrees. The junction blocks that are considered for different configurations by potential merge operations are 3 blocks for the leaf (with 7, 3, and two junction segments in each block respectively) and one block in the octopus (with 3 junction segments).

(c) The optimal setting for the leaf sustains all smaller degree junctions whereas in the octopus it is achieved by merging the whole block into one 8 degree junction segment.

Figure 9: Analyzing blocks of junctions produced by the extended CSP for possible merge operations. A comparison between 9(a) and 9(c) show how the skeleton produced by our method is more successful in reflecting shape properties such as symmetry.

Figure 10: Similar shapes with different topologies and terminal segments.

connecting chords (see highlighted chords in Fig. 9(b)). Each chord has the possibility to be included in the final configuration. If it is not included, it means that the segments on either of its sides are merged to one. We generate all possible configurations, compute the total *strain* measure for each, and select the one with minimal *strain*. While the minimal *strain* setting in the leaf is achieved by sustaining all chords, it requires merging the junction block in the octopus to one 8 degree junction segment (see Fig. 9(b)).
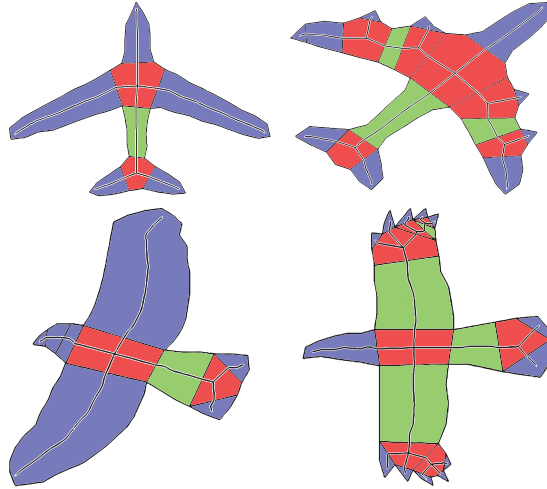
Furthermore, since the role of a junction is more topological than geometric, we try to reduce its size by trimming away triangles from the CDT that, if removed, neither change the topology nor produce more *strain*.

4. The Topological Structure of CAT Segments

Although insignificant terminal parts are pruned, the remaining ones still differ in their level of importance as to what identifies a shape. In Fig. 10 we show similar objects that have different sets of terminal segments. A part–based matching in such cases is expected to match the wing terminal segment in one object to the set of all segments that make up the wing in the other. In order to allow this, without losing detailed features when present, we seek a hierarchical topology where levels correspond to levels of detail.

We first describe the topological structure and connectivity of the segments produced by the CAT skeletonisation. The graphical structure is evident (see Fig. 11). For each segment we associate a node in the graph where terminal, sleeve, and junction nodes have, one, two, and three or more neighbors respectively. Since our data is comprised of silhouettes without holes, the graph connecting these nodes cannot have any cycles.

The relation between features across different levels of detail within the same object is to a certain extent, a node/subgraph relation. For example, while the feathers on the wing in Fig. 10 are terminal nodes, the wing can be represented as a subgraph that includes these nodes. However, not any subgraph is a feature in a higher level of detail representation of the object.
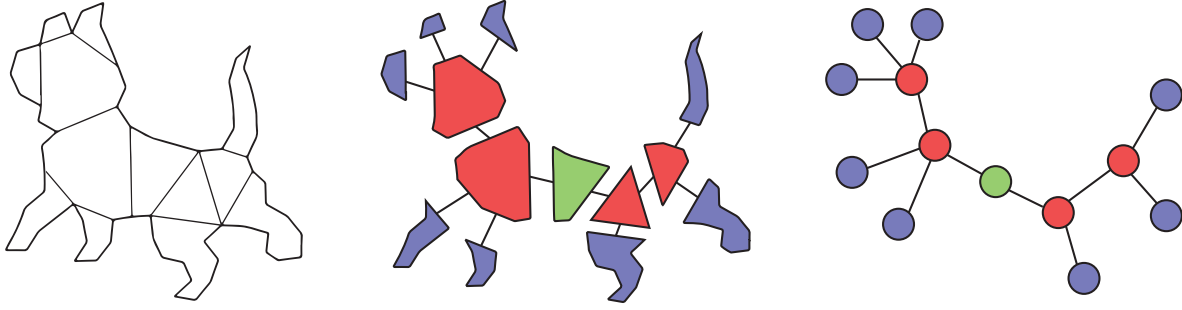
Figure 11: The topology of the CAT segments resulting in a graph structure.

A tree structure of the CAT nodes may resolve this issue provided that the hierarchy is based on geometric as well as topological properties. It is quite obvious that the leaves of a tree representation are the terminal nodes. However, tracing the leaves back to a root is not quite as simple. Our scheme is based on simulated merge operations involving the less salient terminals with their neighborhoods in a recursive manner. At every merging stage, a new set of *compound* terminals emerge. These terminals are comprised of nodes constituting subgraphs that represent potential features.

4.1. Finding the root

The tree construction is based on the node depth computation. Since the node connectivity is already there, it suffices to find a root and the tree structure will follow. The root is expected to have a central position in the shape. So, in addition to the topology of the nodes, their geometry has to be incorporated in the depth computation process. We start from the terminal nodes and assign them zero height values. In an iterative process, smaller terminals are merged with their connecting nodes. For example, in the first iteration, all terminals with area and perimeter values less than σ, a percentage of the whole shape, are merged with their neighbors. Consequently larger terminals are formed and the next iteration merges terminals smaller than $\sigma \times 2$. When all the terminal neighbors of an interior node (sleeve or junction) are merged, it becomes a terminal if it has only one non–terminal neighbor. In this case, it is assigned a height value equal to the index of the current iteration. The remaining one interior node is the root (see Algorithm 1).

Data: sigma
Result: Only one internal node: the root.
while *number of interior nodes > 1* **do**
    forall the *interior nodes N* **do**
        forall the *terminal neighbors T* **do**
            if *areaOfT < sigma AND perimeterOfT < sigma* **then**
                merge T into N;
                NeighborsOfN = NeighborsOfN - 1;
            end
        end
        if *NeighborsOfN = 1* **then**
            change N into a terminal;
        end
    end
    sigma = sigma $\times$ 2
end

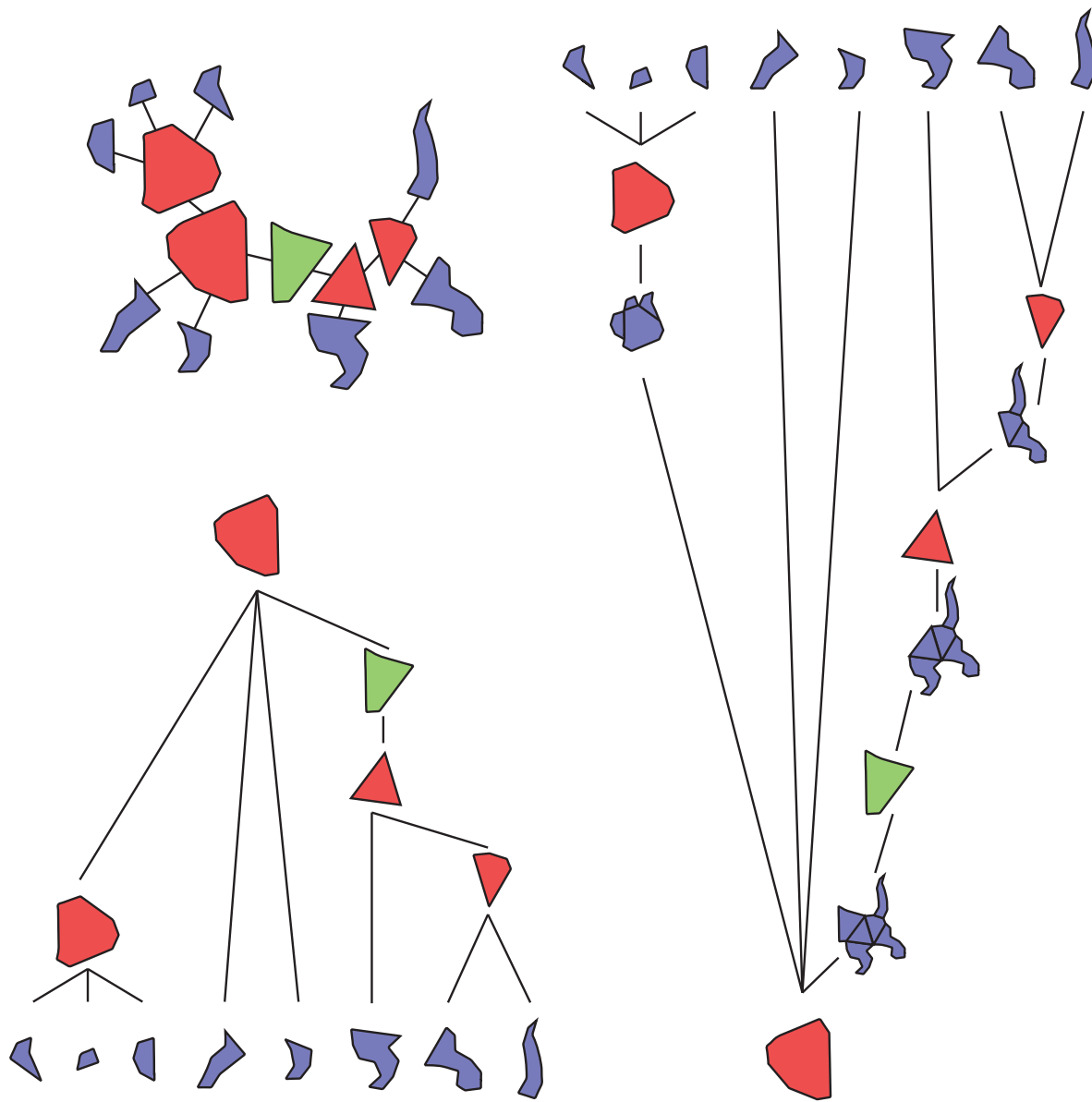**Algorithm 1:** Locating the root of the tree.

14

Figure 12: The bottom up tree shows the iterative process that assigns different levels of depth to junction nodes. The *wings* (compound terminals) build up as the process proceeds starting at the terminals towards the sought root. The top-down tree shows the hierarchical structure of the CAT nodes.

This virtual merge operation yields a new entity which we will use the term 'wing' to refer to. The perimeter and area threshold is increased at every iteration and the remaining interior nodes are tested for merging again (see Fig. 12). The last remaining junction note is considered the root.

4.2. Visual parts' representation in the tree

Visual parts, i.e. the distinctive features in the silhouette of a shape, are not strictly the terminal nodes. This is a consequence of the level of detail of the shape at hand where terminals are the visual parts in the finest level. In Fig. 13 we show two similar objects but which are represented by trees with different topologies. The figure shows two differently sized and placed subtrees that abstractly represent the same part in these objects. So, to have all possible visual parts contribute in the matching process, we need to include some subtrees in addition to the leaves.

However, the size and complexity of the visual parts must be limited so as to preserve the part–based characteristic of the method. To that effect, and as a heuristic measure, we avoid having more than two *salient* features in a visual part. Salience is a visual property related to the part's size and eccentricity. We will introduce the set of conditions that determine the salience of a visual part in the next section after we introduce the geometric parameters describing it.

Each subtree viable for representing a visual part is collected in a new entity that we call a *wing node* (see Fig. 14). The wing construction process resembles that which finds the root. Subtrees with smaller sizes are tested first. If the number of salient visual parts that a subtree contains is equal to or less than 2, a new wing node is formed and added to the set of visual parts representing the shape.

5. The Shape Descriptor

The shape is described by its sequence of visual parts represented as terminal or wing nodes. Each visual part occupies a fragment in the boundary of the silhouette. It is described by a set of shape parameters expressing visual characteristics such as size, degree of protrusion, and eccentricity. Some of these parameters are only derived from the geometry of the terminal parts whereas others are measures taken relative to the whole shape. In addition to the set of numeric values computed for visual parts are:

1. The percentage of its **area**.
2. The percentage of **perimeter** it occupies along the boundary.
3. The **eccentricity** is the ratio of its height away from the shape to the ratio of its width along the shape. In order to compute this, we obtain the minimal bounding box using the rotating calipers algorithm around its convex hull. Then we project the edges of the shape segment on the two sides of the minimal rectangle. The side with the greater sum of projections is the one selected to be aligned with the shape. (see Fig. 15)
4. The **circularity** defined by $4\pi \times p^2/a$, p and a being perimeter and area respectively.
5. **Rectangularity** is the ratio of the area of the segment to that of its minimal bounding rectangle.
6. **Convexity** is the ratio of the perimeter of the convex hull to that of the shape segment. The more convex the shape the higher its convexity. To give a greater weight to objects with lower convexity, we reversed the formula as follows:

$$Convexity_{(reversed)} = (1 - Convexity) \times N_{shape}/N_{hull}$$

where N_{shape} is the number of vertices of the node and N_{hull} is the number of vertices of its convex hull polygon.
7. **Solidity** is the ratio of the area of the shape segment to that of the convex hull. Similar to convexity:

$$Solidity_{(reversed)} = (1 - Solidity) \times N_{shape}/N_{hull}$$

Figure 13: The depth of the interior nodes starting from 0 at the root and increasing as nodes get topologically away from it. Interior nodes are arranged according to depth which is color coded from zero to 6 (horizontal legend). Terminals are color coded according to their clockwise order (vertical legend). The yellow boxes mark subtrees corresponding to equivalent parts in different objects.

Figure 14: Subtrees that are geometrically accepted for wing node generation.

perpendicular side

aligned side

Figure 15: The minimum bounding boxes of terminal nodes. Note that terminals A and B have similar bounding boxes, but the eccentricity of A is less than 1 whereas that of B is closer to 3.

8. **Bending energy** is the sum of curvature values of each point on the boundary segment of this node. To compute the curvature, the whole object is scaled by the reciprocal of the maximal distance to the objects centroid. The discrete curvature at a boundary point B preceded by A and followed by C is:
$$\frac{|(B-A) \times (C-B)|}{|B-A||C-B||A-C|}$$

9. The **chord length ratio** (CLR) is the ratio of the length of the chord connecting the node to the main object to the perimeter of its boundary segment.

10. The radial distance signature of re-sampled points. We re-sample the boundary parts of polygonal segments into N_s points and compute the distance to the segments centroid. We then divide this signature by the unit length which is the ratio of the perimeter of the segment by its number of points.

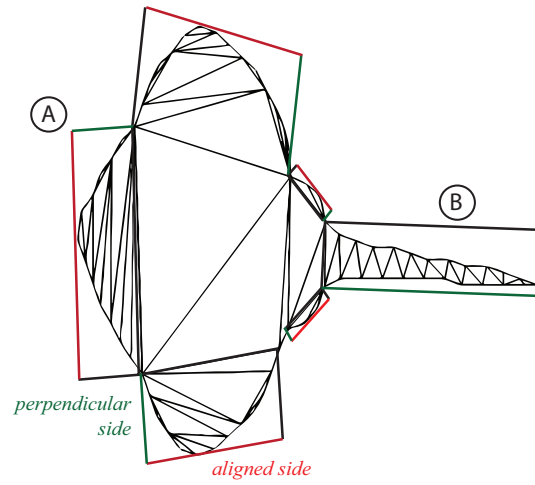In addition to their role in part to part dissimilarity measure, these parameters serve two other purposes. They are used to estimate visual saliency in the wing nodes construction phase. They also constitute the penalty added to the matching cost when the associated visual part is excluded from the correspondence. In what follows we detail out the formulae that evaluate the distance between two visual parts, the penalty, and visual salience assessment.

5.1. The distance between visual parts

Each terminal or wing node is described by two vectors: the assembly of the parameters p and the radial distance signature r. The distance between any two nodes is given by the following formula:

$$d(v_1, v_2) = sqrt\left[\sum_{I=AREA}^{CLR}(p_1[I] - p_2[I])^2\right] + \sum_{i=1}^{N_s}(r_1[i] - r_2[i])^2 \tag{1}$$

Similarly, the penalty of excluding a node is given by:

$$sqrt\left[\sum_{I=AREA}^{CLR}p[I]^2\right] + \sum_{i=1}^{N_s}r[i]^2 \tag{2}$$

The visual parts are not independent of their context in the object. To represent this context, an inter-distance matrix (iDM) saves proximity and angular relations between nodes. Every entry (i,j) in this matrix is a 3 dimensional vector where:

1. The first dimension is the ratio of the Euclidian distance between the end point of node i and the start point of node j to the perimeter of the object.
2. The second dimension is the percentage of the bending energy along the boundary segment between nodes i and j.
3. For every node, we compute the direction vector which is the average of the vectors connecting the midpoint of its chord to the points on the boundary. The third dimension is the angle between the direction vector of node i and node j.

The contribution of inter-distances in the computation of the cost of the match between two shapes involves 4 nodes. For every consecutive pairs of visual parts in the correspondence between shapes A and B, (v_i^A, v_s^B) and (v_j^A, v_t^B), the squared Euclidian distance between $iDM^A(i,j)$ and $iDM^B(s,t)$ is added to the total cost.

5.2. Part salience

During the wing node formation process, if the subtree contains more than two nodes with any salient feature property, the subtree is discarded. We assess the saliency of a visual part by the following tests:
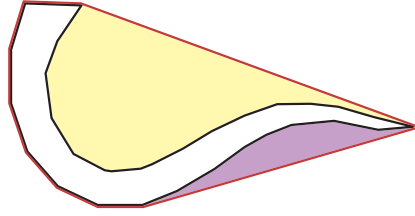
Figure 16: The pockets of void areas between a shape and its convex hull (shaded with color).

1. Relatively **large** size: Area or perimeter percentages greater than a threshold th_{large}
2. Relatively **long** shaped terminals: For perimeter values smaller than th_{large}, if multiplying the perimeter by the eccentricity of the visual node is larger than k_{long}, then it is considered *long* node.
3. **Bent** terminals: We estimate this for perimeter greater than th_{bent} and convexity less than c_{bent}. If the connected pockets of void areas created between the convex hull's perimeter and the terminal's perimeter have areas larger than th_{large} (see Fig. 16), we consider that the feature is salient enough.

6. Matching Shapes

We employ dynamic time warping [26] to find a minimal distance correspondence between parts of matched shapes. Roughly, the idea is to alternatively rotate one shape and fix the other while calculating a distance matrix for every obtained alignment (Algorithm 2). The rotation is topological, i.e. it is performed by shifting the starting point of the terminal parts. The valid *start points* are terminals that are the first to appear in constituting wing nodes along the counter clockwise direction (Algorithm 3). The number of possible *start points* in an object is analogous to the number of skeletal branches connected to its root. Each row in the matrix is the distance between a terminal in the first shape and all terminals in the other. A minimal distance path is calculated for every matrix resulting in one-to-one, one-to-many, many-to-one, or many-to-many pairing. The matrix that produces the minimal distance among others represents the best alignment. We show some examples in Fig. 17 of the first 11 retrievals and the associated correspondence of terminal segments in each match.

6.1. Parameters

The 2D shape databases that we use are sets of binary images. We extract each object's contour and sample it to 100 points for triangulation. Adaptive sampling where the number of sample points is deducted from the shape as in [28] can be applied. However, since the subsequent triangulation will affect skeleton generation, a uniform sampling of the boundary is more preferable. The visual parts' radial distance signature is taken on 12 uniformly sampled points on each part. The geometric attributes presented in section 4 have varying ranges of values. While the area and perimeter are percentage values , the convexity or solidity are less than 1. This gives much more weight in the distance computation to larger valued parameters. To amend this, we apply weights and balance their values into more reasonable ranges. For the same reason, we apply weights to the energy and angle components (second two dimensions) of the vectors of the iDM. These weights are shown in Table 1. Some parameters mentioned through out the paper are listed in Table 1. These values were deducted by inspecting the dataset where terminals with areas less than ϵ (1%) look insignificant and those with areas greater then th_{large} (20%) of the shape look relatively large. As will be shown in the next section, these parameters do not affect the retrieval results as much as they affect the timing performance.

Data: Shape A with n terminals and Shape B with m terminals
Result: A pairing between the visual parts of the shapes and a cost estimate for the match.
minCost = infinity; **forall the** *terminal nodes T in A* **do**

    if *T NOT valid start point (Algorithm 3)* **then**
        | continue;
    end
    rotate the list of terminals so that T is the first;
    Algorithm 4: setup an n × m matrix of Options ;
    for *i = n; i > 0; i–* **do**
        for *j = m; j > 0; j–* **do**
            | Algorithm 5: select the minCostOption at entry (i, j);
        end
    end
    if *minCost > minCost at entry (1,1)* **then**
        | minCost = minCost at entry (1,1);
    end
end
/*Now minCost contains the estimated cost of the optimal correspondence between shapes A and B*/
return minCost;

Algorithm 2: A rough pseudocode describing the dynamic time warping method that finds a minimal cost correspondence between the visual parts of two shapes A and B.

Data: Terminal node TNode
Result: Boolean indicating if this terminal is a valid *start point*.
forall the *Wing nodes W containing TNode* **do**

    if *TNode NOT first terminal of W* **then**
        | return false;
    end
end
return true;

Algorithm 3: Pseudocode that determines if a terminal is a valid *start point*. Terminals are always ordered in their anti-clock direction appearance along the boundary of the shape.

Data: Shape A with n terminals and Shape B with m terminals

Result: A matrix of options where every entry contains the options of matching terminal to terminal, terminal to wing, wing to terminal and wing to wing.

forall the *entry i, j in the matrix* **do**

    add option pair: terminal i of A and terminal j of B;

    forall the *wing nodes w of terminal i* **do**

      | add option pair: wing w of A and terminal j of B;

    end

    forall the *wing nodes w of terminal j* **do**

      | add option pair: wing w of B and terminal i of A;

    end

    forall the *wing nodes w1 of terminal i* **do**

      forall the *wing nodes w2 of terminal j* **do**

        | add option pair: wing w1 of A and wing w2 of B;

      end

    end

    //options to discard each terminal:

    add option: penalty(terminal i in A);

    add option: penalty(terminal j in B);

end

/*An "Option" contains information about matching two nodes ($node_{Option}^A$ and $node_{Option}^B$)from the two shapes. It also saves information about the disjoint previous and next entries in the matrix. This information is essential to avoid having intersecting wings in the resultant correspondence*/

Algorithm 4: The pseudocode that sets up the matrix of options

Data: A list of options at entry (i, j).

Result: The minimal cost option "minCostOption" at this entry. This code also collects a *path* of pairs that constitutes the correspondence in the block [(i, j), (n, m)] of the matrix of options where (n×m) are the dimension sizes of the matrix.

minCostOption = -1;

minCost = infinity;

forall the *Option O at entry i, j* **do**

    // the pair of nodes of O are $node_O^A$ from shape A and $node_O^B$ from shape B;

    fetch Prev-i and Prev-j;//subsequent disjoint entry in the matrix

    subOption = minCostOption in (Prev-i, Prev-j);

    cost of O = distance($node_O^A$, $node_O^B$) + cost of subOption;

    cost of O = cost of O + $(iDM^A(node_O^A, node_{subOption}^A) - iDM^B(node_O^B, node_{subOption}^B))^2$

    if *cost of O < minCost* **then**

      minCost = cost Of O;

      minCostOption = O;

      path to O = ($node_O^A$, $node_O^B$) $\bigcup$ *path* to subOption;

    end

end

/*The set of pairs of the minimal cost correspondence are found in the *path* to minCostOption at entry (1,1) in the matrix of options.*/

Algorithm 5: A rough pseudocode describing the optimal option selection at entry (i, j) of the matrix of options. Details related to the code that prevents intersections between wings that are selected into the correspondence are dropped to make the code more readable.
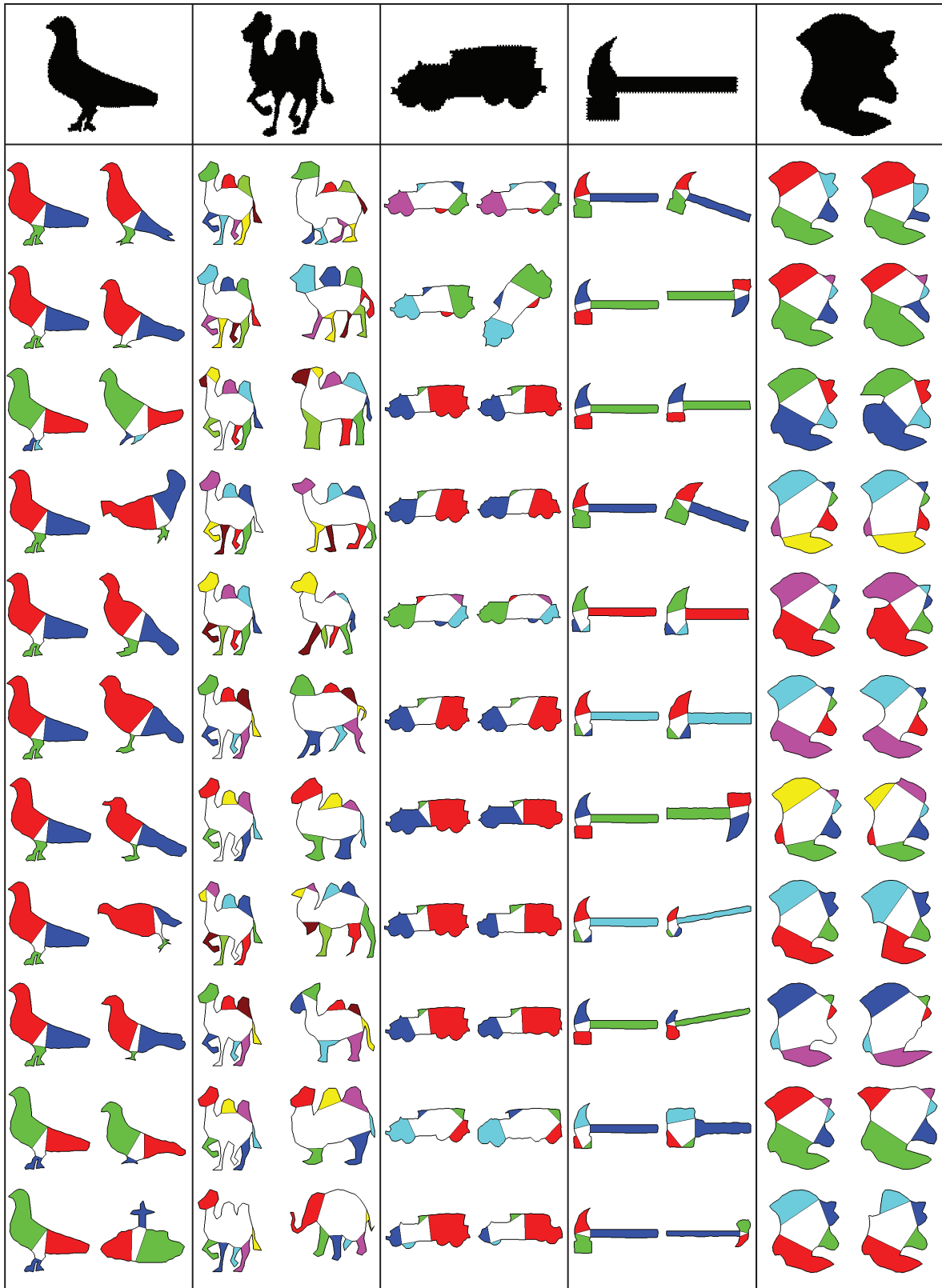
Figure 17: Sample retrieval results on Kimia-216. The top row is the query shape. Each match is illustrated by the query object and the retrieved one where pairs of corresponding terminal segments are shaded by the same color.

23

circularity	40	eccentricity	10	chord length ratio	10
convexity	25	rectangularity	40	energy of iDM	100
solidity	25	bending energy	0.5	angle (in degrees) of iDM	0.1

Table 1: The parameters' and distances' weights.

ϵ	1%	th_{large}	20%	k_{long}	10
μ_0	0.75	th_{bent}	7%	c_{bent}	0.9
μ_1	0.9				

Table 2: Threshold values

6.2. Complexity

The feature vector of each terminal or wing node is made of 22 numeric values representing the parameters and the signature. Table 3 shows statistics about the average number of terminals and wings per object using both the original CAT [29] and the extended CAT on the datasets that we used. The object is also described by the inter-distance matrix which is a 3 dimensional matrix of size: $(N_T + N_W) \times (N_T + N_W) \times 3$ where N_T and N_W are the numbers of terminal and wing nodes respectively.

The matching algorithm constructs a matrix for every *start point* in the two matched objects. The dimensions of each matrix is $N_T^1 \times N_T^2$ where N_T^1 and N_T^2 are the numbers of terminal nodes of the objects. Every entry in the matrix is responsible for computing the cost of all possible matches relating the associated terminals. The number of options at each entry is analogous to the number of wings the terminals belong to. Table 3 shows the average number of wings of the terminals belonging to all the objects in the testing datasets. Having said that, the total number of node distance computations as specified in Eq. 1 is:

$$(N_{start}^1 + N_{start}^2) \times (N_T^1 \times N_T^2) \times (N_{WoT}^i \times N_{WoT}^j)$$

where N_{start} is the number of *start point* of an object and N_{WoT}^i is the number of wings associated with the ith terminal. As such, the time complexity of the whole algorithm is $O(N_{start}^2 \times N_T^2 \times N_{WoT}^2)$.

The number of wings of a terminal N_{WoT} is regulated by the condition on the maximal number of salient parts allowed in a wing. Table 3 shows how the average values of N_{WoT} in both datasets and for both original and extended CAT are roughly the same. This is an indication that we may consider the N_{WoT}^2) factor in the time complexity notation as a constant. N_{start} is analogous to the number of major parts of a shape. It is class related and independent of the level of detail of the silhouette. On the other hand, the finer the level of detail the greater N_T becomes. Since all objects in our system are resampled to a fixed number of points, even N_T becomes under control. Moreover, the average number of *start points* N_{start} and terminals N_T in the extended CAT and those of the original CAT [29] (see Table 3) show that our pruning and merging operations achieved a more than five times faster average retrieval time for both dataset.

6.3. Results

We tested our algorithm on two benchmark databases: Kimia–99 and Kimia–216 [33]. The percentages obtained for the datasets are 98.18 % and 95.83% for the nearest 10 and 11 retrievals respectively. To show the effectiveness of our algorithm in the presence of occlusion, we selected examples from Kimia–99 well known for such cases and showed their retrieval results in Fig. 18. Comparisons of retrieval rates for algorithms tested on these sets are shown in tables 4 and 5. Our method is positioned first on Kimia–216 and obtains the second best score on Kimia–99 dataset. The results show that our method outperforms the skeleton–based methods proposed in [4, 17, 33, 38]. In addition, there is considerable improvement over using the segmentation of the original CAT.

The effectiveness of each part of the feature vector in the distance estimation can be made sense of by a few experiments. We first set all parameter weights to zero. The average precision for Kimia–99

	Terminals (N_T)	Wings (N_W)	Start points (N_{start})	Wings of Terminal (N_{WoT})
Extended CAT	6.91	8.81	4.68	3.66
Original CAT	9.00	19.49	8.06	3.70
Extended CAT	7.19	9.54	5.29	3.45
Original CAT	10.44	18.65	8.70	3.60

Table 3: Mean values of the factors that affect the complexity of our algorithm. The first two rows are for Kimia-99 and rows 3 and 4 are for Kimia-216 dataset.

and Kimia–216 are 92.5% and 90.5% respectively. When all weights are set to 1, the obtained results are 93.7% and 90.6%. Turning the computations around by discarding the radial distance signature from the feature vector and resetting the weights to those in Table 1, the results drop to 90.7% and 86.9% for the two datasets.

The weights of the geometric parameters (eccentricity, circularity, bending energy, rectangularity, convexity, solidity, and chord length ratio) are tuned to balance their effect with respect to each other and improve the retrieval results. Setting the weight of each parameter to zero in turn, we rerun the experiments to obtain the average precision for each dataset. The 9 precision values for Kimia–99 and Kimia–216 are in a close range around 95% and 92% respectively. Moreover, the precision recall plots for the 9 runs show how the parameters with their current weights have a somewhat similar effect on the retrieval results.

Method	1st	2nd	3rd	4th	5th	6th	7th	8th	9th	10th	Total
Shape context [6]	97	91	88	85	84	77	75	66	56	37	76.36
Shock graph [33]	99	99	99	98	98	97	96	95	93	82	96.56
Path similarity [4]	99	99	99	99	96	97	95	93	89	73	94.84
Symbolic representation [11]	**99**	**99**	**99**	**98**	**99**	**98**	**98**	**95**	**96**	**94**	**98.48**
MGVF [17]	99	99	99	99	99	99	99	98	90	87	97.77
Two strategies [38]	99	99	99	98	99	99	99	97	96	84	97.87
Height functions [42]	99	99	99	99	98	99	99	96	95	88	98.08
Our method with the original CAT	98	95	95	90	89	84	78	67	66	50	82.02
Our method with the extended CAT	**99**	**99**	**99**	**99**	**99**	**99**	**99**	**98**	**96**	**85**	**98.18**

Table 4: The precision table for Kimia-99 dataset. It is made of 9 classes with 11 objects each. The total retrieval result is 98.18% in the nearest 10 matches. The highest two retrieval results are highlight in boldface.

7. Conclusion and Future Work

We propose a new shape descriptor inspired by human vision and cognition. Our descriptor is comprised of the arrangement of visually salient parts along its boundary. Based on an extended version of the chordal axis transform, we segment the shape and embed the segmentation in a hierarchical structure. This allows capturing parts on different levels of detail. Based on a part representation by radial distance signature, and using a set of simple geometric properties, we devised a dynamic programming approach to find a minimal cost correspondence between two shapes. We show that a redundance-prone representation of overlapping boundary fragments can be avoided. Balanced on
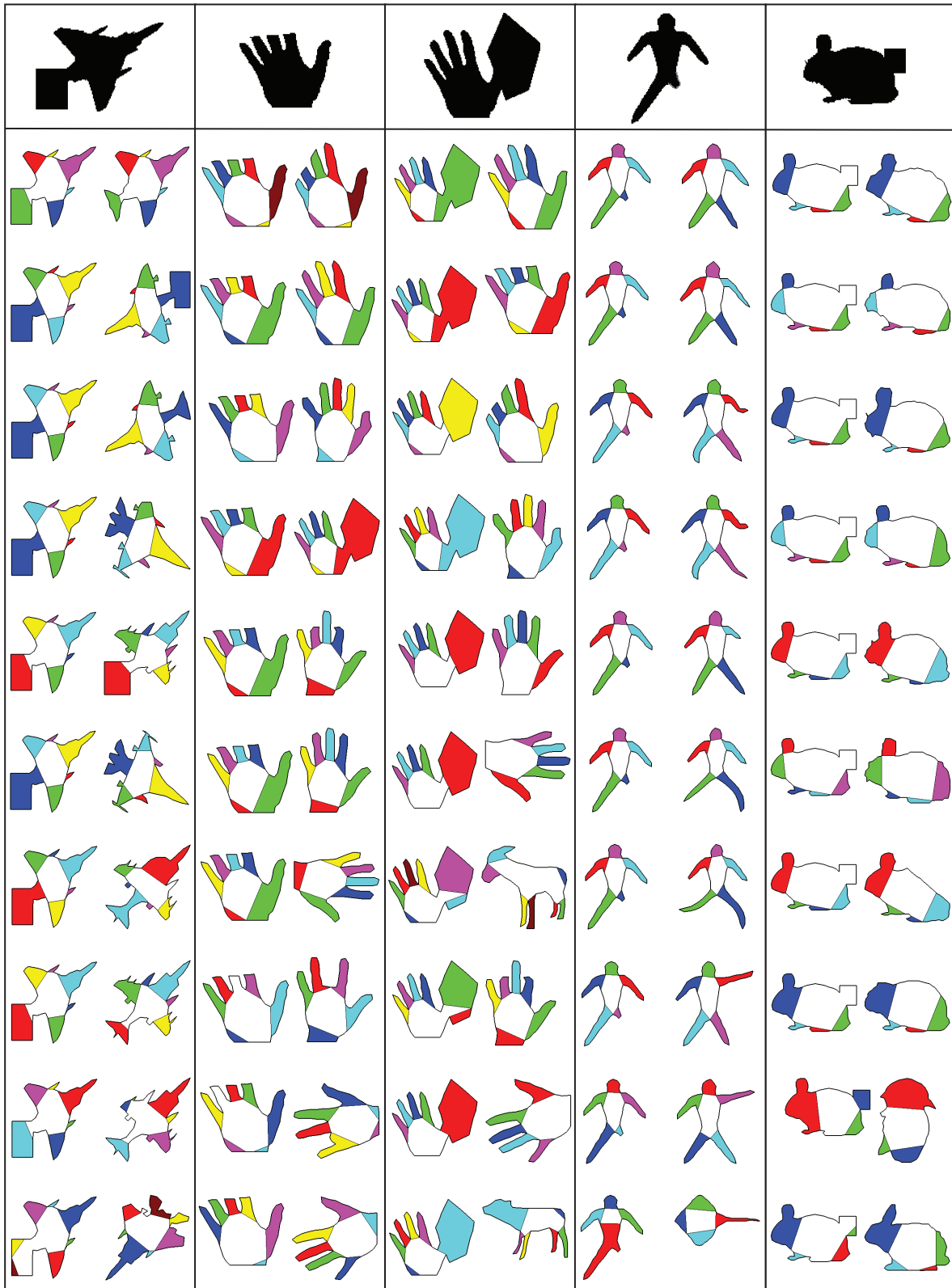
Figure 18: The retrieval results of objects selected from Kimia-99 with occluded parts. Each match is illustrated by the query object and the retrieved one where pairs of corresponding terminal segments are shaded by the same color.

Method	1st	2nd	3rd	4th	5th	6th	7th	8th	9th	10th	11th	Total
Shape context [6]	214	209	205	197	191	178	161	144	131	101	78	76.13
Shock graph [33]	216	216	216	215	210	210	207	204	200	187	163	94.44
MGVF [17]	216	216	216	216	216	214	211	209	205	187	160	95.37
Path similarity [4]	**216**	**216**	215	**216**	213	210	210	207	205	191	**177**	**95.79**
Our method with the original CAT	214	215	211	207	200	198	190	178	164	158	123	86.61
Our method with the extended CAT	**216**	**216**	**216**	214	**215**	**212**	**213**	**208**	**204**	**195**	**168**	**95.83**

Table 5: The precision table for Kimia-216 dataset. It is made of 18 classes with 12 objects each. The total retrieval result is 95.83% in the nearest 11 matches. The highest two retrieval results are highlight in boldface.
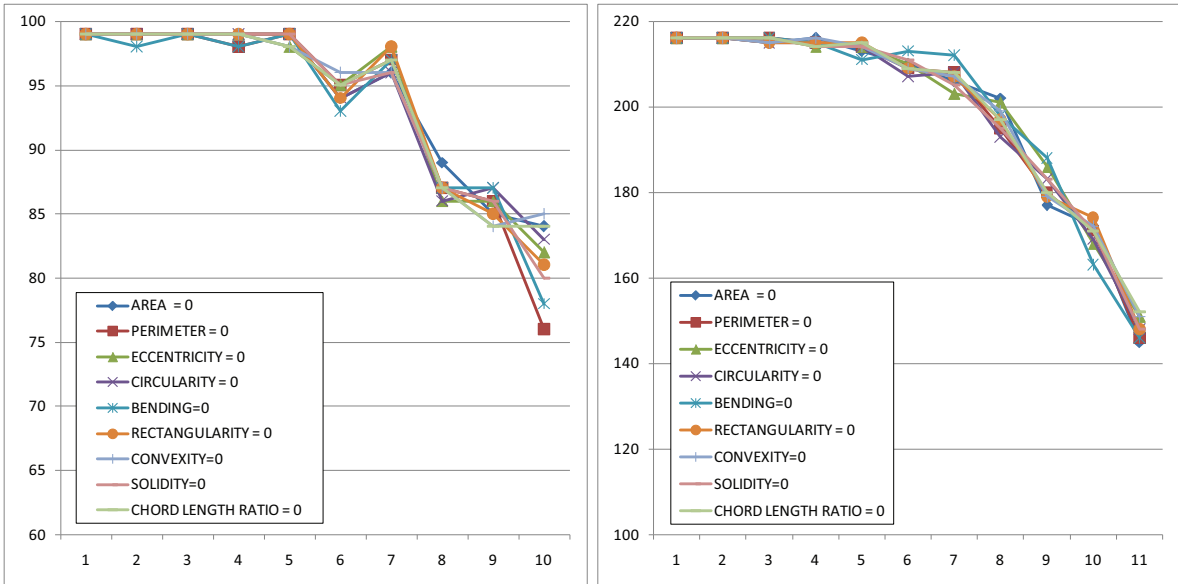

Figure 19: The precision recall plots of Kimia–99 (right) and Kimia–216 (left) obtained by setting the weight of each parameter to zero one at a time. The graphs show how the weights distribute the effect of the parameters uniformly.

rather simple building blocks of shape representation and matching, our approach produces good retrieval results on benchmark databases.

Having assessed the validity of our method on 2D image datasets, our future direction is to apply it on sketch–based 3D object retrieval. Generalizing this part correspondence matching to 3D [41] is an interesting problem. A possible direction for future work is to build a classifier based on the codebook of visual parts learnt from matching objects of the same class. A similar idea has been proposed by Kim et al. [21] in the context of 3D objects.

8. Acknowledgements

References

[1] Cagri Aslan, Aykut Erdem, Erkut Erdem, and Sibel Tari. Disconnected skeleton: Shape at its absolute scale. *IEEE Trans. Pattern Anal. Mach. Intell.*, 30(12):2188–2203, December 2008.

[2] Xiang Bai, Longin Jan Latecki, and Wen-Yu Liu. Skeleton pruning by contour partitioning with discrete curve evolution. *IEEE Trans. Pattern Anal. Mach. Intell.*, 29(3):449–462, March 2007.

[3] Xiang Bai, Wenyu Liu, and Zhuowen Tu. Integrating contour and skeleton for shape classification. In *Computer Vision Workshops (ICCV Workshops), 2009 IEEE 12th International Conference on*, pages 360–367. IEEE, 2009.

[4] Xiang Bai, Xingwei Yang, Deguang Yu, and Longin Jan Latecki. Skeleton-based shape classification using path similarity. *IJPRAI*, 22(4):733–746, 2008.

[5] D. H. Ballard and C. M. Brown. *Computer Vision*. Prentice-Hall, Englewood Cliffs, NJ, 1982.

[6] S. Belongie, J. Malik, and J. Puzicha. Shape matching and object recognition using shape contexts. *IEEE Trans. Pattern Anal. Mach. Intell.*, 24(4):509–522, April 2002.

[7] Marco Bertamini and Johan Wagemans. Processing convexity and concavity along a 2-d contour: figureground, structural shape, and attention. *Psychonomic Bulletin & Review*, 20(2):191–207, 2013.

[8] Irving Biederman. Recognition-by-components: A theory of human image understanding. *Psychological Review*, 94:115–147, 1987.

[9] Carolyn Backer Cave and Stephen M. Kosslyn. The role of parts and spatial relations in object identification. *Perception*, 22(2):229–248, 1993.

[10] Elias H. Cohen and Manish Singh. Geometric determinants of shape segmentation: Tests using segment identification. *Vision Research*, 47(22):2825 – 2840, 2007.

[11] Mohammad Reza Daliri and Vincent Torre. Robust symbolic representation for shape recognition and retrieval. *Pattern Recognition*, 41(5):1782–1798, 2008.

[12] Mohammad Reza Daliri and Vincent Torre. Classification of silhouettes using contour fragments. *Computer Vision and Image Understanding*, 113(9):1017 – 1025, 2009.

[13] Arnaud Delorme, Ghislaine Richard, and Michele Fabre-Thorpe. Key visual features for rapid categorization of animals in natural scenes. *Frontiers in Psychology*, 1(21), 2010.

[14] Aykut Erdem and Sibel Tari. A similarity-based approach for shape classification using aslan skeletons. *Pattern Recogn. Lett.*, 31(13):2024–2032, October 2010.

[15] Jacob Feldman, Manish Singh, Erica Briscoe, Vicky Froyen, Seha Kim, and John Wilder. *An Integrated Bayesian Approach to Shape Representation and Perceptual Organization*, pages 55–70. Advances in Computer Vision and Pattern Recognition. Springer London, 2013.

[16] Ngo Truong GIANG, Ngo Quoc TAOb, and Nguyen Duc DUNG. Skeleton-based shape matching using higher-order constraints. *Journal of Pattern Recognition & Image Processing*, 4(4):443–454, 2013.

[17] Wooi-Boon Goh. Strategies for shape matching using skeletons. *Computer Vision and Image Understanding*, 110(3):326 – 345, 2008. Similarity Matching in Computer Vision and Multimedia.

[18] Ge Guo, Yizhou Wang, Tingting Jiang, Alan L. Yuille, and Wen Gao. Computing importance of 2d contour parts by reconstructability. In *ICCV Workshops*, pages 1364–1371. IEEE, 2011.

[19] I.V. Higgins and S.M. Stringer. The role of independent motion in object segmentation in the ventral visual stream: Learning to recognise the separate parts of the body. *Vision Research*, 51(6):553 – 562, 2011.

[20] D.D. Hoffman and W.A. Richards. Parts of recognition. *Cognition*, 18(13):65 – 96, 1984.

[21] Vladimir G Kim, Wilmot Li, Niloy J Mitra, Siddhartha Chaudhuri, Stephen DiVerdi, and Thomas Funkhouser. Learning part-based templates from large collections of 3d shapes. *ACM Transactions on Graphics (TOG)*, 32(4):70, 2013.

[22] Nacra Laiche, Slimane Larabi, Farouk Ladraa, and Abdelnour Khadraoui. Curve normalization for shape retrieval. *Signal Processing: Image Communication*, 29(4):556 – 571, 2014.

[23] Longin Jan Latecki, Rolf Lakamper, and Diedrich Wolter. Shape similarity and visual parts. In *In Proceedings of the 11th International Conference on Disrecte Geometry for Computer Imagery (DGCI*, pages 34–51, 2003.

[24] Cong Lin and Chi-Man Pun. Shape classification using hybrid regional and global descriptor. *International Journal of Machine Learning & Computing*, 4(1), 2014.

[25] Tianyang Ma and Longin Jan Latecki. From partial shape matching through local deformation to robust global shape similarity for object detection. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 1441–1448. IEEE, 2011.

[26] Cory Myers, Lawrence Rabiner, and Aaron E Rosenberg. Performance tradeoffs in dynamic time warping algorithms for isolated word recognition. *Acoustics, Speech and Signal Processing, IEEE Transactions on*, 28(6):623–635, 1980.

[27] Peter Neri. Wholes and subparts in visual processing of human agency. *Proceedings of the Royal Society B: Biological Sciences*, 276(1658):861–869, 2009.

[28] Dilip K Prasad, Maylor KH Leung, Chai Quek, and Siu-Yeung Cho. A novel framework for making dominant point detection methods non-parametric. *Image and Vision Computing*, 30(11):843–859, 2012.

[29] Lakshman Prasad. Rectification of the chordal axis transform skeleton and criteria for shape decomposition. *Image and Vision Computing*, 25(10):1557 – 1571, 2007. Discrete Geometry for Computer Imagery 2005.

[30] Rahul Rai and Akshay V Deshpande. Fragmentary shape recognition: A bci study. *Computer-Aided Design*, 71:51–64, 2016.

[31] Thomas Sebastian, Philip Klein, and Benjamin Kimia. Recognition of shapes by editing shock graphs. In *Computer Vision, IEEE International Conference on*, volume 1, pages 755–755. IEEE Computer Society, 2001.

[32] Thomas B. Sebastian, Philip N. Klein, and Benjamin B. Kimia. Shock-based indexing into large shape databases. In *Computer Vision–ECCV 2002*, pages 731–746. Springer, 2002.

[33] Thomas B. Sebastian, Philip N. Klein, and Benjamin B. Kimia. Recognition of shapes by editing their shock graphs. *IEEE Trans. Pattern Anal. Mach. Intell.*, 26(5):550–571, May 2004.

[34] Kaleem Siddiqi, Ali Shokoufandeh, Sven J Dickinson, and Steven W Zucker. Shock graphs and shape matching. *International Journal of Computer Vision*, 35(1):13–32, 1999.

[35] Manish Singh. *Oxford Handbook of Perceptual Organization*. Oxford University Press, USA, 1 edition, August 2015.

[36] Kang B. Sun and Boaz J. Super. Classification of contour shapes using class segment sets. In *CVPR (2)*, pages 727–733. IEEE Computer Society, 2005.

[37] Hari Sundar, Deborah Silver, Nikhil Gagvani, and Sven Dickinson. Skeleton based shape matching and retrieval. In *Shape Modeling International, 2003*, pages 130–139. IEEE, 2003.

[38] Andrew Temlyakov, Brent C. Munsell, Jarrell W. Waggoner, and Song Wang. Two perceptually motivated strategies for shape classification. In *CVPR*, pages 2289–2296. IEEE, 2010.

[39] Andrea Torsello and Edwin R. Hancock. A skeletal measure of 2d shape similarity. *Comput. Vis. Image Underst.*, 95(1):1–29, July 2004.

[40] Andrea Torsello, Antonio Robles-Kelly, and Edwin R. Hancock. Discovering shape classes using tree edit-distance and pairwise clustering. *Int. J. Comput. Vision*, 72(3):259–285, May 2007.

[41] Oliver Van Kaick, Hao Zhang, Ghassan Hamarneh, and Daniel Cohen-Or. A survey on shape correspondence. In *Computer Graphics Forum*, volume 30, pages 1681–1707. Wiley Online Library, 2011.

[42] Junwei Wang, Xiang Bai, Xinge You, Wenyu Liu, and Longin Jan Latecki. Shape matching and classification using height functions. *Pattern Recogn. Lett.*, 33(2):134–143, January 2012.

[43] Xinggang Wang, Bin Feng, Xiang Bai, Wenyu Liu, and Longin Jan Latecki. Bag of contour fragments for robust shape classification. *Pattern Recognition*, 47(6):2116 – 2125, 2014.

[44] Joeri De Winter and Johan Wagemans. Segmentation of object outlines into parts: A large-scale integrative study. *Cognition*, 99(3):275 – 325, 2006.

[45] Jun Xie, Pheng-Ann Heng, and Mubarak Shah. Shape matching and modeling using skeletal context. *Pattern Recognition*, 41(5):1756 – 1767, 2008.

[46] Mingqiang Yang, Kidiyo Kpalma, and Joseph Ronsin. A survey of shape feature extraction techniques. *Pattern recognition*, pages 43–90, 2008.