A Visibility Graph based Shape Decomposition Technique

Foteini Fotopoulou and Emmanouil Z. Psarakis
Department of Computer Engineering & Informatics, University of Patras, Rio-Patras, Greece

Keywords: Shape Decomposition, Visibility Graph, Shared Neighbors.

Abstract: In this paper, a new shape decomposition method named Visibility Shape Decomposition (VSD) is presented. Inspired from an idealization of the visibility matrix having a block diagonal form, the definition of a neighborhood based visibility graph is proposed and a two step iterative algorithm for its transformation into a block diagonal form, that can be used for a visually meaningful decomposition of the candidate shape, is presented. Although the proposed technique is applied to shapes of the MPEG7 database, it can be extended to 3D objects. The preliminary results we have obtained are promising.

1 INTRODUCTION

Shape decomposition constitutes a vital procedure in the field of computer vision, that is able to distinguish the different components of the original object and split it into meaningfull components. Meaningfull components are defined as parts that can be perceptually distinguished from the remaining object. Dividing a shape into its meaningful subparts has gained much attention as it is useful in many applications including, but not limited to, shape compression, shape retrieval, shape analysis and shape matching. Moreover, it is well known that decomposition assists recognition of shapes with occluded or deleted parts, inversely proportional growth of several parts, moving parts etc. Thus, the decomposition of an arbitrary shape into its components constitutes a problem of great interest.

In this paper the shape decomposition problem is addressed and a novel decomposition technique, referred to here as Visibility Shape Decomposition (VSD) is proposed. The introduced method uses the well established idea of visibility (De Berg et al., 1997) among shape's contour points and appropriately achieves to organize boundary points into a block-diagonal matrix, thus obtaining the shape decomposition into meaningful components. More precisely, a two step algorithm that efficiently exploits the visibility information and results in organizing the boundary points into meaningful components is proposed.

Let us consider a *plane curve*, that describes a shape boundary, to be defined from the path traced by the following N position vectors:

$$\mathbf{r}(i) = (x(i),\ y(i)),\ i = 1, 2, \cdots, N. \quad (1)$$

Then, we can form the following Visibility Graph $\mathcal{G}_V = (\mathcal{V}, \mathcal{E}, \mathcal{W})$, where $\mathcal{V}, \mathcal{E}, \mathcal{W}$ are the nodes and edges sets and a binary weighted matrix respectively. More precisely, in this graph model of the *plane curve*, nodes' set $\mathcal{V}$ is defined as follows:

$$\mathcal{V} = \{\mathbf{r}(i),\ i = 1, 2, \cdots, N\}, \quad (2)$$

and the w_{ij} element of the $N \times N$ matrix $\mathcal{W}$ can be defined as follows:

$$w_{ij} = \begin{cases} 1, & \text{if nodes } i, j \text{ are visible} \\ 0, & \text{otherwise} \end{cases} \quad (3)$$

where nodes $i,\ j$ are considered as visible if, the following *Visibility Rules* hold:

- $\mathcal{VR}_1$: The connecting edge ε_{ij} does not intersect with the *plane curve*.
- $\mathcal{VR}_2$: The connecting edge ε_{ij} is totally located inside the *plane curve*.

The visibility between two points has been widely studied in motion plan and computational geometry (De Berg et al., 1997). Although in a 2D shape the visibility between its boundary points can be easily checked, an appropriate extension can be made for 3D objects (Liu et al., 2011).

The introduced method originates from the idea of visibility between boundary points i, j as it is defined in Equ. (3). Obviously, the construction of such a visibility graph $\mathcal{G}_V$ is invariant to rigid deformations, such as shape rotation, translation and scaling, since they do not affect neither the nodes nor the

Fotopoulou F. and Z. Psarakis E..
A Visibility Graph based Shape Decomposition Technique.
DOI: 10.5220/0004692005150522
In *Proceedings of the 9th International Conference on Computer Vision Theory and Applications* (VISAPP-2014), pages 515-522
ISBN: 978-989-758-003-1

edges positioning. The $\mathcal{G}_V$ of the camel (Figure 1(a)), obtained by the application of the above mentioned *Visibility Rules*, is depicted in Figure 1(b). As it is obvious the structure of this matrix does not facilitate shape partitioning. An ideal matrix for shape decomposition would have the form of a block-diagonal similarity matrix, where its non-overlapping blocks could represent the shape's parts, in a sequential manner as it is shown in Figure 1(c). The basic idea behind this idealization is that each shape component can be represented by a block in the respective block-diagonal matrix[1]. Thus, the shape decomposition problem can be restated as follows: Given the visibility matrix $\mathcal{W}$ construct a block diagonal matrix that best approximates the desired form.

In this work we propose a technique called VSD for solving the above defined problem. In order to achieve our goal the n-conditioned $\mathcal{G}_V$ is introduced and a two-step iterative algorithm is proposed for transforming the original $\mathcal{G}_V$ into the desired block-diagonal form. Namely, in the first step of the proposed algorithm we attempt to "extend" the near field visibility by using a principle similar to the "shared neighboors" according to which the similarity of two nodes is confirmed by the number of their Shared Nearest Neighbors (Jarvis and Patrick, 1973). This procedure results in a Voting graph which constitutes the focal matrix of our decomposition scheme. In the second stage, an appropriate thresholding procedure takes place which efficiently converts the above mentioned matrix into a crisp one that is used as input in the next iteration of the algorithm. This iterative compactification procedure results in a block diagonal matrix, which best approximates the idealization form and additionally delivers the structure of the original shape appropriately organized into meaningful groups.

Although the introduced decomposition method does not apply the two major perception rules, i.e. the minima rule (Singh et al., 1999b) and the short cut rule (Singh et al., 1999a), the resulting shape clustering often corresponds well to decompositions that a human might make. Comparisons with the state of art methods validate that our method produces satisfactory results.

The remaining of the paper is organized as follows: In Section 2 a literature overview is presented. In Section 3 the introduced method is explained in detail and in Section 4 experimental results are shown. Finally the paper concludes in Section 5.

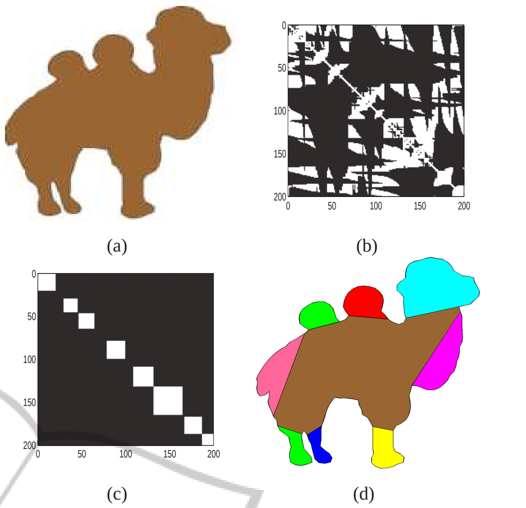


Figure 1: The camel-shape (a). The corresponding $\mathcal{G}_V$ (b). An ideal block diagonal matrix for the camel shape decomposition (c). Each block in the diagonal representation denotes a specific shape component (d).

2 RELATED WORK

Shape segmentation is important in perceptual understanding of objects and facilitates recognition that is robust to occlusions, movements, deletion or growth of portions of an object (Biederman et al., 1987), (Siddiqi and Kimia, 1995). Shape decomposition techniques, can be broadly classified into two main categories. In the first one methods that aim to partition a shape into its meaningful parts belong, while in the second shapes are segmented usually by following a certain geometric rule.

Determination of a shape part as natural is not obvious due to the involvement of the human perception. However, there exist two major rules that exploit the results of the cognitive theory over the human's identification of a shape. In (Singh et al., 1999b) a set of cuts is defined, such that their endpoints denote negative minima of curvature of the corresponding boundary. Continuing, in (Singh et al., 1999a) -which complements the minima rule- cuts are selected according to their length.

In (Latecki and Lakämper, 1999) a 2D shape is decomposed into its convex parts, at different evolution steps, using a rule similar to minima rule. In addition, the method proposed in (Mi and DeCarlo, 2007) is specially designed for decomposing a shape into its meaningful parts. In (Juengling and Mitchell, 2007) the minima and short-cut rules are used for shape de-

[1]In order the ideal matrix to be block-diagonal, we should start the decomposition from the beginning of a group in the shape boundary. Otherwise, the first block of the ideal matrix is divided into four sub-blocks covering the four corners of the matrix. However, by taking into account their similarity and for simplicity purposes, we do not distinguish between them.

composition. In particular, a constrained Delaunay triangulation of the shape is computed and an optimum set of cuts (i.e. interior edges) is selected, by solving a combinatorial optimization problem.

On the other hand, methods that decompose a shape into certain geometric models usually refer to convex parts decomposition. However, decomposition of a shape to strictly convex parts, often results in an uncontrollable number of segments. Consequently, approximate convex parts are used instead. In (Lien and Amato, 2004) polygons are decomposed in an hierarchical way, by iteratively removing the most significant non-convex parts. Authors of this paper indicate the facility of computing approximate convex segments opposed to exactly convex ones. In (Liu et al., 2010) candidate cuts are computed by Morse theory and their total length is minimized. In order to avoid production of redundant parts while resulting in natural shape decomposition, in (Ren et al., 2011) the selection of the best set of candidate cuts that has both a minimum size and a high visual naturalness is proposed. The latter property is obtained by imposing the minima rule and the short cut rule. In (Siddiqi and Kimia, 1995) it is reported that convex parts depict in some way the human's conception towards decomposition activities. Therefore, both (Liu et al., 2010) , (Ren et al., 2011) fall into both main shape decomposition categories and they aim to result in meaningful shape partitioning.

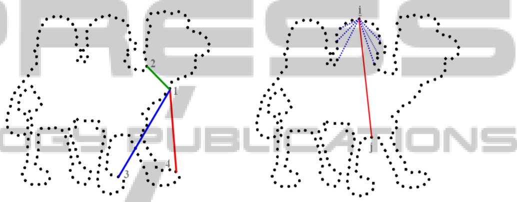
Other methods (De Goes et al., 2008) exploit the multiscale properties of diffusion distance, in an attempt to achieve robust shape decomposition in articulated objects. In (Shapiro and Haralick, 1979) a graph theoretic clustering method is introduced where a 2D shape is partitioned into clusters that intuitively correspond to shape parts. Our method is based on a clustering technique where as we are going to see clusters' similarity is assessed according to an appropriate voting procedure.

3 THE VSD METHOD

Let us consider that the visibility matrix mentioned in Section 1, is given. Then, the proposed method aims at appropriately transforming the original visibility matrix into a block diagonal one, which can be easily used for the visually meaningful decomposition of the candidate shape. However, as it was already mentioned, the form of the original visibility matrix is not appropriate to serve our goal, and must be properly transformed. This is exactly the goal of the following subsection.

3.1 Neighborhood Based Visibility

A shape can be easily transformed into a graph, where the boundary points stand for the graph's nodes and the edges are the lines connecting those nodes. However, such a graph has no physical significance, due to the fact that many nodes are connected even though they do not "see" each other. Assuming an observer placed at each node, a more natural result could be obtained by linking the nodes that fall within the observer's field of view. According to the *Visibility Rules* of Section 1, a node pair is defined as visible if its corresponding edge does not intersect with the contour and locates inside the area enclosed by the contour. It is clearly shown in Figure 2(a) where nodes 1, 2 form a visible pair, while nodes 1,3 and nodes 1,4 are not visible.



(a) (b)

Figure 2: According to *Visibility Rules*, nodes 1,2 are visible, while nodes 1,3 and 1,4 are not visible (a). Despite nodes i, j are visible they do not contribute to shape decomposition as the nodes that are linked by the dotted lines (b).

Despite naturalness in the shape representation that is introduced by the visible node pairs, there still exist edges that do not help in distinguishing the shape's parts, as we can see in Figure 2(b) where the nodes i, j form a visible pair according to the above mentioned *Visibility Rules*. However, we are interested in revealing the meaningful parts of the shape, which means that nodes i, j do not contribute to that way as they link nodes from different shape parts. Although nodes i, j can "see" each other, they are not close neighbor points (starting from the i boundary point and moving clock-wise). These connections manifest themselves in Figure 3(a) as the non-zero values that lay away from the main diagonal thus making the solution of the shape partitioning problem complicated. On the other hand, in Figure 2(b), nodes that are linked with i showed on dotted edges, are found in the same neighborhood with it and form the camel's hump. Therefore, $\mathcal{G}_V$ must be redefined by posing nodes' neighborhood restriction. Specifically, two boundary points are defined as not visible and their corresponding edge is set to zero, if they are placed far from each other. Notice that distance is not measured according to the edge length but is calculated as the number of intermediate points when moving along the contour. Another interesting approach could use edge length information, but is not studied in this paper.

Let us concentrate ourselves on the definition of the conditioned visibility graph. To this end, let us define the neighborhood of radius n of the i-th node of set $\mathcal{V}$ defined in Equ. (2) as the following set of indices:

$$\mathcal{N}_n^i = \{< i+j >_N,\ j = 0, \pm 1, \pm 2, \dots \pm n\} \qquad (4)$$

where $< . >_N$ denotes the modulo N operation, that is:

$$0 \leq\ < k >_N = k - mN\ \leq\ N-1,\ m \in \mathcal{Z}. \qquad (5)$$

As it can be easily verified, the range of valid values of radius n belongs to the set $\mathcal{R}_I = \{1, 2, \cdots, \lceil \frac{N}{2} \rceil\}$, where $\lceil x \rceil$ denotes the ceil of x. Finally, we must stress at this point that the use of the modulo operator is essential for the correct definition of the neighborhood of a node, because the shape contours are closed curves.

Having defined the radius-n neighborhood of a node of $\mathcal{G}_V$, the elements of the conditioned weighted matrix $\mathcal{W}^n$ can be defined as follows:

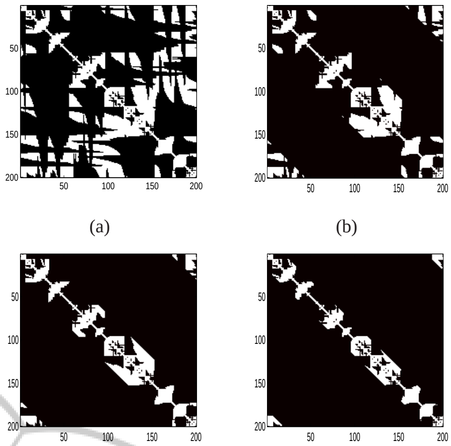
$$w_{ij}^n = \begin{cases} w_{ij}, & \text{if } j \in \mathcal{N}_n^i \\ 0, & \text{otherwise.} \end{cases} \qquad (6)$$

Note that the conditioned weighted matrix defined above can easily be formed from the element-wise multiplication of the original weighted matrix $\mathcal{W}$ and the following symmetric band Toeplitz matrix:

$$T_n = toeplitz([\mathbf{1}_{n+1}\ \mathbf{0}_{N-(2n+1)}\ \mathbf{1}_n]) \qquad (7)$$

where $\mathbf{1}_k$, $\mathbf{0}_k$ are vectors of length k containing all ones and zeros, respectively. Note finally that the l_0 “norm” of each row of the above formed matrix $\mathcal{W}^n$ is at most $2n+1$.

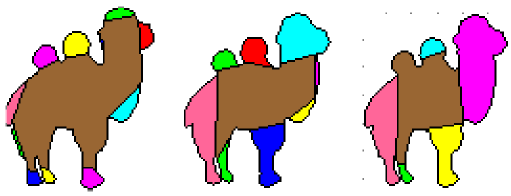
Conditioned visibility graphs resulting from the original visibility graph of the camel shape, depicted in Figure 3(a), are shown in Figure 3(b)-3(d) for three (3) different values of radius n. As it is clear from this figure, the constraint imposed through the radius of the neighborhood of each boundary point to the field of view of the hypothetical observer, influences the size, as well as, the specific form of the shape components that emerge at a great degree. More specifically, if the admissible number of neighboring nodes that can be visible is small, then more yet smaller groups of the boundary nodes have the tendency to be formed (see Figure 3). On the other hand, imposing a loose constraint to $\mathcal{G}_V$, groups have the tendency to expand, leading, in most cases, to undesirable results. Note also from Figure 3(a) that total absence of such a visibility constraint results to a $\mathcal{G}_V$ with non distinct components, while edges that link nodes of physically different groups are present.



(a) (b) (c) (d)

Figure 3: Initial $\mathcal{G}_V$ (a) and n-conditioned $\mathcal{G}_V$ for different values of n: 40 (b), 30 (c) and 10 (d).

We are expecting that different values of neighborhood radius, result in different decompositions of the candidate shape. Indeed, as a rule, the greater the radius of the neighborhood is, the larger the size of the shape's components are. This phenomenon, is evident in Figure 4, where the results obtained from the application of the proposed decomposition algorithm (presented in the next subsection) in the conditioned visibility graphs depicted in Figure 3(b)-3(d) corresponding to different values of n, are shown.



(a) $n = 10$ (b) $n = 30$ (c) $n = 40$

Figure 4: Different decompositions of the camel shape corresponding to different values of neighborhood radius n (see text).

It is obvious that for small values of n small shape components are constructed, but as n increases the small components seem to merge and larger groups are formed. As it becomes clear, we must somehow estimate the desired radius of neighborhood of the graph.

Moreover, although in the n-conditioned $\mathcal{G}_V$ several groups of boundary points seem to be formed more clearly, as it is evident in Figure 3, a strict limitation seems to be imposed by the *Visibility Rules* $\mathcal{VR}_i$, $i = 1, 2$. Indeed, very often in the n-conditioned $\mathcal{G}_V$ values of boundary points are set to zero, since according to this rule they are not visible. However, in many cases, the majority of these nodes belong to the same part of the shape. Such a case is shown, as an example, in Figure 5 where, as it is apparent, all nodes i, j, k and m are members of the "camel's head" part. Nevertheless, the boundary points i, j cannot "see" each other, due to the intersection existing between their corresponding edge and the shape's contour, which means that the i-j link gains no connecting value. Application of such a strict rule leaves out many important points from each shape component (i.e. nodes i, j in Figure 5), thus leading into a n-conditioned $\mathcal{G}_V$ where groups are not clearly separated and in general its form is far away from the idealization presented in Section 1.

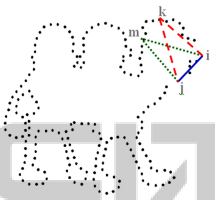


Figure 5: Despite violation of the rule $\mathcal{VR}_2$, nodes i, j are gaining connection votes as they are both visible from nodes k and m.

In an attempt to overcome this problem and define the shape's meaningful parts, as we are going to see in the next section, we assign to each node a number that indicates connection votes. A connection vote is assigned to a link each time the corresponding nodes are both visible from another node. In Figure 5, nodes i, j both form visible pairs with nodes k and m, and thus two (2) votes are added to the edge ε_{ij}. In that way, visibility is extended and many edges are obtaining"visibility votes" even if they are not visible. This "indirect" visibility facilitates the connection of nodes that belong to the same group, even though they do not all share direct links. The number of votes that an edge accumulates is analogous to the number of nodes from which the edge's ending nodes are mutually visible. This alternative way to define similarity between the node-pairs is similar, but does not coincide, to the "Shared Nearest Neighbors" algorithm, which was first introduced by (Jarvis and Patrick, 1973). According to this principle the similarity of the connected nodes i, j which are both connected to a set of nodes $\mathcal{A}$, is augmented as their closeness is "confirmed" by nodes of set $\mathcal{A}$.

Having identified the critical problems that drastically affect the quality of the solution of the problem at hand, we are going to investigate them in detail in the next subsection.

3.2 Neighborhood Radius Estimation

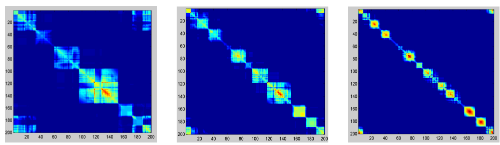
As mentioned the impact of the size of the neighborhood, to the quality of the obtained shape decomposition, in terms of the visual meaningfulness of its components, is essential. Thus its determination, even in a non optimal way, is considered necessary. To this end, let us consider that the idealization $\mathcal{W}^{n^\star}$ introduced in Section 1, of the visibility matrix is given, and let us form the following sequence of numbers:

$$c[n] = \sum_{i=1}^{N} \sum_{j \in \mathcal{N}_n^i} v_{ij}^n, \; n = 1, 2, \cdots, \lceil \frac{N}{2} \rceil \qquad (8)$$

where v_{ij}^n denotes the $(i,\ j)$ element of the following "Voting" matrix:

$$V^n = \mathcal{W}^n (\mathcal{W}^n)^T = (\mathcal{W}^n)^2. \qquad (9)$$

Note that by taking into account the special "$1-0$" form of the components of its basic ingredient, that is the elements of matrix $\mathcal{W}^n$, the content of the $(i,\ j)$-th element of the above defined voting matrix, expresses the connectivity strength of the nodes i and j in the shape contour. The voting matrices of the corresponding n-conditioned $\mathcal{G}_V$ shown in Figures 3(b), 3(c) and 3(d), are depicted in Figure 6, for neighborhood radius n=40, 30 and 10, respectively. Note that their form is closer to the idealization introduced in Section 1. This contitutes the basic reason we propose its use as a vehicle to fulfill our ultimate goal, as we are going to see in the next subsection. We have to say more things about this voting matrix, but for the moment let us concentrate ourselves on the "Mean Voting" sequence defined in Equ. (8), and in particular on a very interesting property that it has and we are going to exploit in order to estimate the desired value $n^\star$.



(a) (b) (c)

Figure 6: Examples of Voting Matrices corresponding to different values of neiborhood radius n (see text).

More precisely, using the definition of sets $\mathcal{N}_n^i$, we can easily see that the cardinalities of any pair of the aforementioned sets satisfy the following ordering relations:

$$|\mathcal{N}_{n+k}^i| \geq |\mathcal{N}_n^i|, \; \forall \; k \geq 0 \qquad (10)$$

and consequently, as it can easily be proven, $c[n]$ is an increasing sequence of the neighborhood radius, that is:

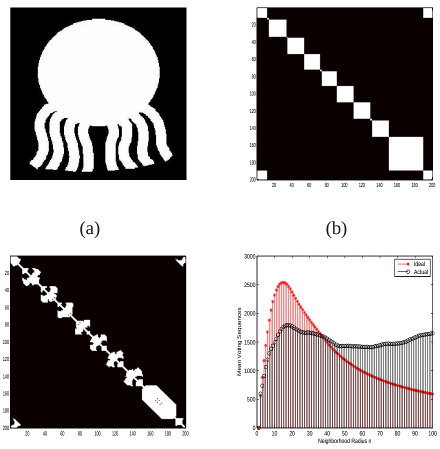
$$c[n+k] \geq c[n], \; \forall \; k \geq 0. \qquad (11)$$

In addition, there is a radius after that the values of the sequence should not be changed even more. As it is clear, that value is the desired $n^\star$.

Thus, considering $\mathcal{W}^{n^\star}$ known, by computing the sequence $c[n],\ n = 1,2,\cdots,\lceil\frac{N}{2}\rceil$ from Equ. (8) and by exploiting the above mentioned properties, the true value of the neighborhood radius can be obtained from the solution of the following optimization problem:

$$n^\star = \arg\max_{n\in\mathcal{R}_I}\left(\frac{c[n]}{2n+1}\right). \qquad (12)$$

However, the ideal matrix $\mathcal{W}^{n^\star}$ is unknown and the computation of the sequence defined in Equ. (8) is practically impossible. Instead, the use of the actual n-conditioned visibility matrices is proposed for its evaluation and then an estimation of the desired neighborhood radius $\hat{n}^\star$ can be achieved by solving the corresponding optimization problem. An example of the above mentioned sequences resulting from the solution of the above mentioned optimization problems to the "octapus" shape is shown in Figure 7. We must stress at this point that even though the exact and the approximated sequences are totally different, the values of the corresponding neighborhood radii, in most cases, are close to each other. Furthermore, the impacts of "dark areas" along the main diagonal as well as the off diagonal "bright areas" appeared in the sequence of the actual matrices are clearly depicted on the form of "Mean Voting Actual" sequence shown in Figure 7(d).



(a) (b) (c) (d)

Figure 7: The octapus shape contained in MPEG7 CE-shape1 part B (a), its ideal block diagonal matrix (b), the actual $\hat{n}^\star$-conditioned $\mathcal{G}_V$ (c) and the corresponding "Mean Voting" sequences (d).

Having estimated the neighborhood radius $\hat{n}^\star$ for a candidate shape, let us go back to "Voting" matrix V^n defined in Equ. (9) and see how we can use it to solve the second critical problem mentioned in the last paragraph of the previous subsection.

3.3 The Iterative Algorithm

What remains in order to complete the presentation of the proposed technique is the development of an algorithm for the solution of the inverse problem, that is, given the actual conditioned visibility matrix $\mathcal{W}^{\hat{n}^\star}$ form a block diagonal matrix with each block representing a meaningful part of the shape contour. To this end, we propose a two step iterative algorithm with its first step being a *normalization*-step of the "Voting" matrix V^n defined in Equ. (9) and the second one a *thresholding*-step. Let us now concentrate on the above mentioned matrix and see how we can use it in order to achieve our goal. As it is clear from its definition in Equ. (9), the $(i,\ j)$ element of the matrix can be expressed by the inner product of the i-th and j-th rows of the matrix $\mathcal{W}^{\hat{n}^\star}$, that is:

$$v_{ij}^{\hat{n}^\star} = <\mathbf{w}_i^{\hat{n}^\star},\ \mathbf{w}_j^{\hat{n}^\star}>. \qquad (13)$$

Using the definition of the inner product, Equ. (13) can be rewritten as follows:

$$v_{ij}^{\hat{n}^\star} = ||\mathbf{w}_i^{\hat{n}^\star}||_2\ ||\mathbf{w}_j^{\hat{n}^\star}||_2 \cos(\theta_{ij}) \qquad (14)$$

where $||\mathbf{x}||_2$ denotes the l_2 norm of visibility vector $\mathbf{x}$, and θ_{ij} the existing angle between the two mentioned visibility vectors.

We can now define the following normalized version of the above mentioned matrix

$$\bar{v}_{ij}^{\hat{n}^\star} = \cos(\theta_{ij}) \qquad (15)$$

with the value of each element, quantifying the existing similarity or correlation of the visibility vectors corresponding to nodes i and j of the $\hat{n}^\star$-conditioned visibility graph.

Note now that after the normalization step, matrix $\bar{V}^{\hat{n}^\star}$ takes values in the interval $[0\ 1]$, and thus our goal now is to update the $\mathcal{W}^{\hat{n}^\star}$ by using a predefined threshold t:

$$\mathcal{W}^{\hat{n}^\star} = \bar{V}^{\hat{n}^\star} \geq t \qquad (16)$$

and use it as the new approximation of the ideal visibility matrix in the next iteration.

The above described steps are repeated in an iterative fashion until the convergence to a block diagonal matrix be achieved.

An outline of the proposed algorithm follows.

procedure SHAPE DECOMPOSITION($\mathcal{V}$, t)
  Using $\mathcal{VR}_i,\ i = 1,2,$ form matrix $\mathcal{W}$
  Form $c[n],\ n = 1,2,\cdots,\lceil\frac{N}{2}\rceil$ of Equ. (8).
  Solve Problem defined in (12) to find $\hat{n}^\star$.
  Form $\mathcal{W}^{\hat{n}^\star}$ using $\mathcal{W}$ and $T_{\hat{n}^\star}$ of Equ. (7).
  repeat
    Form the Normalized Voting matrix $\hat{V}^{\hat{n}^\star}$

using Equ. (15).
Use Equ. (16) to update matrix $\mathcal{W}^{\hat{n}^*}$.
until convergency
end procedure

Having completed the presentation of the proposed technique, in the next section we are going to apply it on several shapes and compare its performance against well known in the literature techniques.

4 EXPERIMENTAL RESULTS

In this section we are going to present comparative results obtained from the application of the proposed technique on several shapes of the MPEG7 CE-shape-1 part B database (Latecki et al., 2000). All shape contours we used in our experiments, were sampled at N=200 points. In addition, in all experiments we have conducted, the value of threshold t was set to 0.5.

A sample of 2D decomposed shapes are shown in Figure 8. It is evident that for some shapes the resulting decomposition is meaningful, while in some situations, parts that obviously could not be separated by a human being, are splitted by the algorithm. Specifically, the crown shapes are decomposed into seven (7) or eight (8) parts, while one could expect a six-part result. Continuing, in the fly shapes legs, wings and antennas are successfully recognized by our method. The same comment holds for the butterflies shown in the fourth row of the figure. Moreover, in shapes such as the deer, the elephants and the dogs, the introduced method achieves partitioning into meaningful components (i.e. legs, head, tails etc). However, in some cases the start and the end boundary points of a component should be defined more explicitly, so as to better demostrate the meaningful component that is captured. Finally, shapes such as the bone or the margaret-like ones, are well partiotioned. Regarding the non-rigid parts of shapes (i.e. legs, tails, proboscis, antennas etc) we can observe that the VSD method provides satisfactory results, as it achieves to recognize almost all meaningful articulated shape parts, even if they are depicted in different poses. Despite the fact that non-rigid deformations affect the visibility graph (different edges are revealed, while others may dissapear), the key idea of connection votes, seems to give the ability to handle non-rigid shape variations, in an effective way.

To further demonstrate the effectiveness of the proposed technique, a small sample of the results we obtained from the application of our technique as well as the ACD (Lien and Amato, 2004), the CSD (Liu et al., 2010) and the MNCD (Ren et al., 2011) methods on shapes contained in the MPEG7 CE-shape-1 part B database, are shown in Figure 9. As we can see from this figure, the proposed method performs well in cases of non-rigid parts. Indeed, the beetle's legs as well as the horse's legs are not decomposed into multiple parts, an indication that the results of the proposed technique are close to perceptually meaningful decompositions. On the other hand, methods that use concavity rules fail to capture such parts as a whole, because of their large concavity. Having no intention to discredit the performance of the other techniques, the VSD fails to recognize the head of the

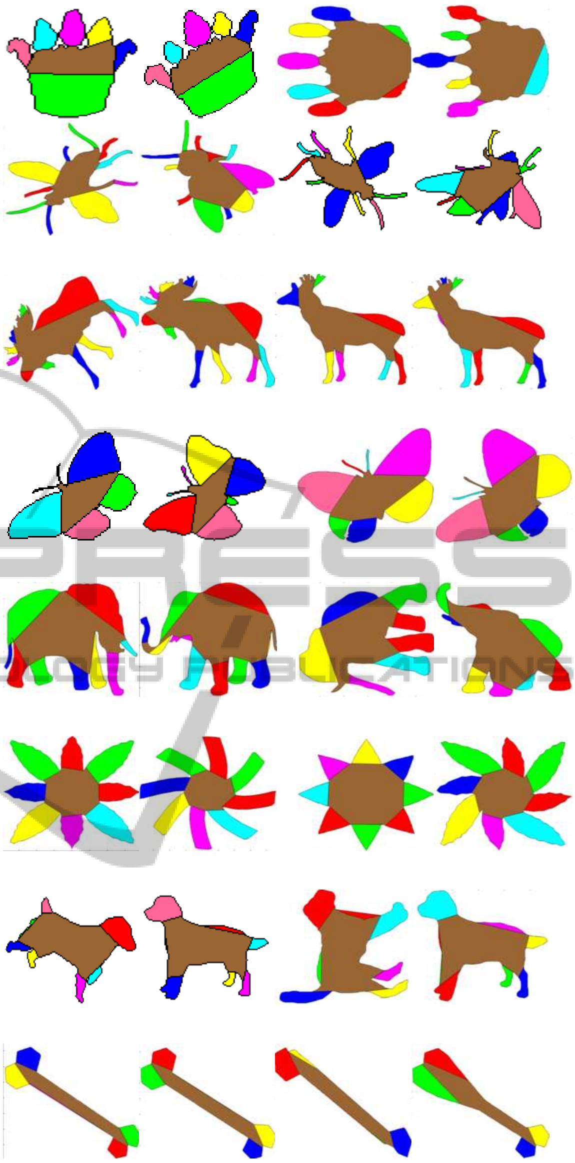


Figure 8: A sample of decomposed shapes of the MPEG7 database.

mouse shape as a whole part due to the constraint that is imposed to the Visibility Graph by the estimated value $\hat{n}^{\star}$ of radius.

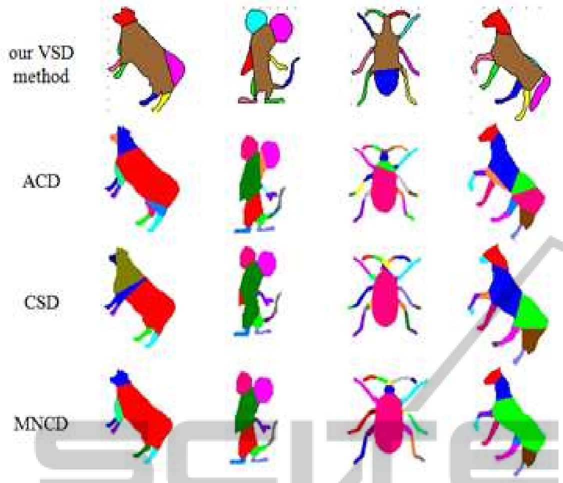


Figure 9: A small sample of decomposition results obtained from the application of our method and ACD, CSD and MNCD methods in shapes contained in MPEG7 database.

5 CONCLUSIONS

In this paper, a novel method for shape decomposition is presented, which stems from the idea of visibility between boundary points. The shape is segmented into its meaningful parts by a two-step algorithm which efficiently organizes shape's boundary points into separate blocks. The proposed method is applied in a large number of 2D shapes. The introduced technique seems to result in perceptualy more meaningful decompositions of the non-rigid parts of the shapes, than most of the existing methods. In addition, the proposed technique results in decompositions that nearly captures the whole structure of the original shape. The extension of the proposed technique in 3D shapes and the development of new robust algorithms to efficiently converge to the ideal low rank shape matrix are currently under investigation.

REFERENCES

Biederman, I. et al. (1987). Recognition-by-components: A theory of human image understanding. *Psychological review*, 94(2):115–147.

De Berg, M., Van Kreveld, M., Overmars, M., and Schwarzkopf, O. (1997). *Computational geometry: algorithms and applications*. Springer.

De Goes, F., Goldenstein, S., and Velho, L. (2008). A hierarchical segmentation of articulated bodies. In *Computer graphics forum*, volume 27, pages 1349–1356. Wiley Online Library.

Jarvis, R. and Patrick, E. (1973). Clustering using a similarity measure based on shared nearest neighbors. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 22(11).

Juengling, R. and Mitchell, M. (2007). Combinatorial shape decomposition. In *Advances in Visual Computing*, pages 183–192. Springer.

Latecki, L. J. and Lakämper, R. (1999). Convexity rule for shape decomposition based on discrete contour evolution. *Computer Vision and Image Understanding*, 73(3):441–454.

Latecki, L. J., Lakamper, R., and Eckhardt, T. (2000). Shape descriptors for non-rigid shapes with a single closed contour. In *Computer Vision and Pattern Recognition, 2000. Proceedings. IEEE Conference on*, volume 1, pages 424–429. IEEE.

Lien, J.-M. and Amato, N. M. (2004). Approximate convex decomposition of polygons. pages 17–26.

Liu, H., Liu, W., and Latecki, L. J. (2010). Convex shape decomposition. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 97–104. IEEE.

Liu, Y.-S., Ramani, K., and Liu, M. (2011). Computing the inner distances of volumetric models for articulated shape description with a visibility graph. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 33(12):2538–2544.

Mi, X. and DeCarlo, D. (2007). Separating parts from 2d shapes using relatability. In *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*, pages 1–8. IEEE.

Ren, Z., Yuan, J., Li, C., and Liu, W. (2011). Minimum near-convex decomposition for robust shape representation. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, pages 303–310. IEEE.

Shapiro, L. G. and Haralick, R. M. (1979). Decomposition of two-dimensional shapes by graph-theoretic clustering. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, (1):10–20.

Siddiqi, K. and Kimia, B. B. (1995). Parts of visual form: Computational aspects. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 17(3):239–251.

Singh, M., Seyranian, G., and D.Hoffman (1999a). Parsing silhouettes: The short-cut rule. *Perception and Psychophysics*, 61(4):636–660.

Singh, M., Seyranian, G. D., and Hoffman, D. D. (1999b). Parsing silhouettes: The short-cut rule. *Perception & Psychophysics*, 61(4):636–660.