

Dense sampling of shape interiors for improved representation

Vittal Premachandran and Ramakrishna Kakarala

School of Computer Engineering, Nanyang Technological University, Singapore

ABSTRACT

Matching shapes accurately is an important requirement in various applications; the most notable of which is object recognition. Precisely matching shapes is a difficult task and is an active area of research in the computer vision community. Most shape matching techniques rely on the contour of the object to provide the object's shape properties. However, we show that using the contour alone cannot help in matching all kinds of shapes. Many objects are recognised because of their overall visual similarity, rather than just their contour properties. In this paper, we assert that modelling the interior properties of the shape can help in extracting this overall visual similarity. We propose a simple way to extract the shape's interior properties. This is done by densely sampling points from within the shape and using it to describe the shape's features. We show that using such an approach provides an effective way to perform matching of shapes that are visually similar to each other, but have vastly different contour properties.

Keywords: Shape Matching, Perceptual Techniques

1. INTRODUCTION

Measuring similarities between objects is an active area of research in computer vision and obtaining precise similarity measures is an unsolved problem. A strong and robust similarity measure can be beneficial in many applications such as shape-matching, shape-retrieval, and shape-based object recognition. However, the lack of an accurate definition of similarity makes the task of obtaining a measure all the more difficult. Most pattern recognition problems are required to overcome the apparent vagueness in the definition of similarity and come up with a quantitative similarity (or dissimilarity) measure between objects. In order to do so, the algorithms have to first extract appropriate features such that the features both well describe the object and allow for a discriminative comparison between objects of different classes. The challenge that remains is to come up with good features that well-describe an object. In this paper, we look at the problem of shape analysis and propose ways to improve current shape-matching techniques.

2-D shape-matching is currently being thought of as matching respective contours of the shapes. A shape, S , is considered to have a well-defined boundary, \mathcal{B} . Most algorithms define features at sampled locations on this boundary. They then solve a correspondence problem to match the two sets of features, and the cost of matching the two sets of features is considered as the cost of matching the respective shapes. While most of the shape information can usually be extracted from just the object's contour, it is not necessarily the case for objects that have a strong base structure. Noise on the boundaries of such shapes have minimal effect on the human visual system.

Figure 1 shows examples of four different object classes, with two instances from each class, where the instances of each object class are visually similar to each other even though some of them have multiple indentations in their contours. As discussed in,¹ people tend to neglect these minor (or even major) indentations while perceiving the object's shape. This is in accordance with Gestalt psychology, which maintains that the human eye sees objects in their entirety before perceiving their individual parts. The gestalt effect is the form-generating capability of our senses, particularly with respect to the visual recognition of figures, and whole forms, instead of just a collection of simple lines and curves. Due to the Gestalt effect, approaches that perform shape-matching based on part decomposition, or curve matching, will not perform well on objects such as those shown in Figure 1. So, there is a need for the development of shape-matching techniques that can somehow capture this Gestalt effect.²

Further author information: (Send correspondence to Vittal Premachandran)

Vittal Premachandran: E-mail: vittalp@pmail.ntu.edu.sg

Ramakrishna Kakarala: E-mail: ramakrishna@ntu.edu.sg

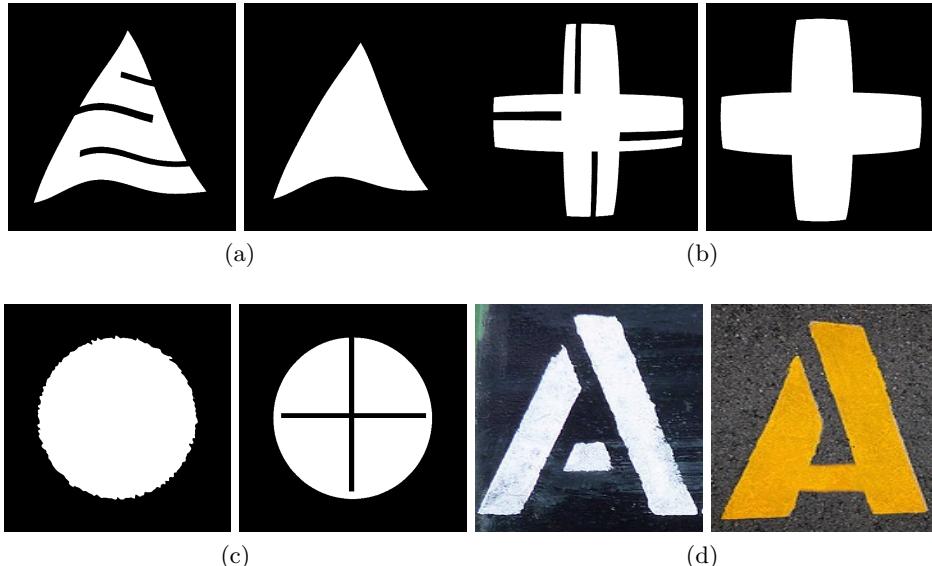


Figure 1: Figure shows examples of objects whose visual similarity does not depend on their contour properties. Visually similar objects have been grouped into subfigures. Shape-matching techniques that perform contour-based matching of shapes will give incorrect matches while matching such objects.

This calls for other shape characteristics to be used while describing a shape, instead of the widely used contour properties. Looking at the shapes in Figure 1, it is clear that the new features have to capture the global shape properties. Global shape properties does not only mean the complete contour, but also the interior layout of the shape. In this paper, we propose a simple, but effective, way to capture the shape’s interior properties. We propose to model the interior properties of the shape by densely sampling points from within the shape. We show that using the interior properties of the shape can help capture the global shape properties better than using the contour. With the use of features obtained from the dense interior points, we will see from experiments in Section 4 that the shape matching procedure gains invariance to indentations in the contour. Portions of this work also appear in another study.¹ Here, we provide some quantitative results that show the benefit of using our proposed sampling method over another frequently-used sampling approach, i.e. the accept/reject technique. We also show the benefits of a dense representation of shapes on 3-D objects by showing improvements in symmetry detection.

2. RELATED WORK

Some notable advances that have been made in the area of shape-matching are discussed below. A typical approach to measure shape similarity is by means of non-rigid shape deformation.^{3,4} Such methods measure the ease, or difficulty, in transforming one shape into another. Most approaches perform shape-matching as a matching of the respective boundaries of the objects. The shape boundaries are divided into a set of n landmark points, $S = \{p_1, p_2, \dots, p_n\}$, for easier representation and matching. Belongie et al.⁵ showed that these points need not be restricted to extremal points on the curve, and could be located at any place on the object boundary. They also proposed the use of shape contexts (SC) at each of these sampled points, to describe the shape. The shape context at each sampled point is represented as a 2-D histogram of distances and angles, which is obtained using the rest of the $n - 1$ points.

Ling et al.⁶ proposed a variant of SC, namely, Inner Distance Shape Context (IDSC), to overcome the problem of articulations in shapes. Instead of using Euclidean distance and angle, as used in SC, the IDSC generates the histograms at the sampled points using inner distance (the length of the shortest path connecting the two points, such that the path lies completely within the shape) and inner angle. The use of inner distance and inner angle makes the descriptor invariant to articulations. Though IDSC looks at distances between points such that the

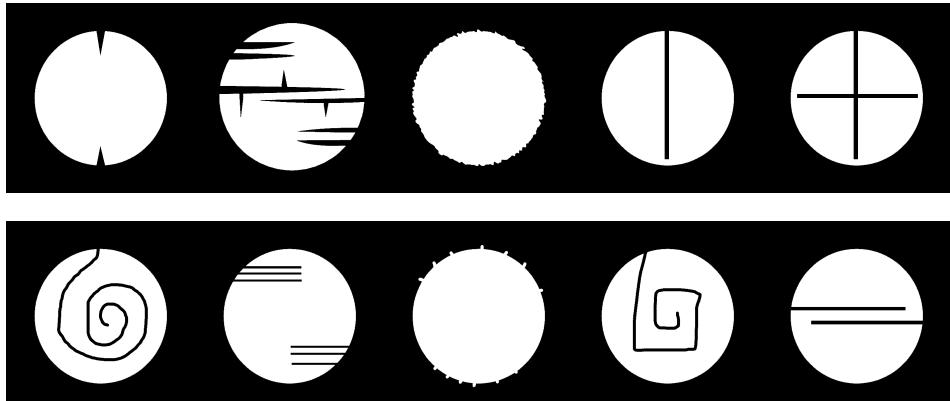


Figure 2: Figure shows examples of a particular class of objects from the MPEG7 database. All of the above objects have different contour properties. However, their overall visual similarity is still that of a circle.

path connecting them lies completely within the shape’s boundary, it still cannot capture the interior density of the shape. Being reliant on the boundary of the shape for calculating the shape context histograms, IDSC will give incorrect matches while matching objects such as those shown in Figure 2.

Though the use of inner distance provided invariance to articulations, it could not be directly applied to “non-ideal” 2-D projections of 3-D objects. If the projection took place using a weak perspective, then not all parts of the 3-D model would get accurately projected onto the 2-D plane. In order to overcome this problem, Gopalan et al.⁷ proposed to perform independent affine normalization of the object’s parts. They consider near convex portions of the object as separate parts of the object. They then used IDSC to perform matching on the normalized shapes. However, if such an approach is followed for shapes such as those shown in Figure 1, objects with indentations will be split into far more number of parts than objects without indentations.

Certain authors^{8,9} have used the Medial Axis Transform (MAT) and its variant, shock graphs, for matching shapes . The medial axis, or skeleton, is the locus of the centers of all maximally inscribed circles of the object. Xie et al.,¹⁰ proposed to model shapes using skeletal contexts. Their contexts are calculated at the ends of the skeleton and the bins are populated by the non-uniformly sampled points from the boundary. The reliance on the boundary of the object to obtain the skeleton, makes these approaches susceptible to noise in the boundary. A shape with indentations will have more branches in the skeleton than a shape without indentations.

Recently, some effort has gone into the development of perceptually motivated techniques. Temlyakov et al.¹¹ propose to split the object into a base structure and multiple strand structures. Base structures and strand structures are compared separately. They use IDSC for comparing base structures and just check if the two objects have similar number of strands. More recently, Hu et al.¹² proposed a morphological approach to model human perceptions. Morphological closing is used to “close the shapes”. To compare the shapes, they use IDSC before and after performing the morphological operation, and retain the better of the two costs.

The common trend that we see in almost all of the above-mentioned works is that they either try to perform shape matching directly using contours, or using features extracted from contours. In the next section, we show how the interior properties of the shape can be modelled by simple approximation techniques. We can then describe features that make use of these interior properties. These new features will help us escape the problems that one faces while relying on the object’s contour.

3. DENSE SAMPLING OF SHAPES

We would like to take the reader back to the examples in Figures 1 and 2. We identify that the human visual system not only recognises shapes by their external contour, but also by their “density”. We perceive a solid disc as a different object compared to a ring, though both have a circle as their outer contour. From this example, we can see that the interior solidity is vital in properly classifying an object. We therefore come up with a simple sampling-based technique to capture the shape density.

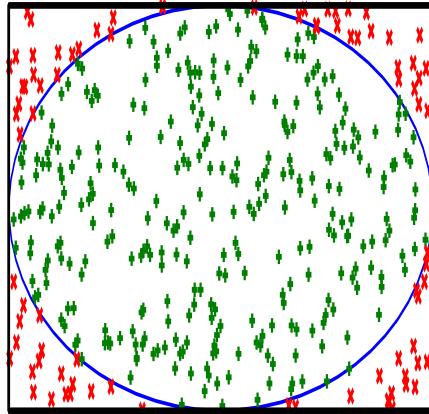


Figure 3: In order to sample points from within a circle, we uniformly sample points from a bounding square. We retain the points that fall within the circle ('+') and reject the points that fall outside the circle ('x').

We propose to approximate the interior shape of an object by sampling points lying within the object's boundary. Each part of the object is equally important in understanding the shape properties. Therefore, we use an uniform sampling scheme to sample points that lie uniformly within the shape. It is easy and straightforward to sample uniformly distributed points from well-known and simple shapes such as a square, rectangle, circle, or a triangle. However, uniformly sampling a fixed set of points from a random shape is not that simple.

A common technique that is usually adopted while sampling is the accept/reject technique. An arbitrary shape is encompassed using a well known, and simple shape (say, a square), and points are sampled uniformly from the enclosing shape. The points that fall within the enclosed shape are retained and the rest of the points are rejected. Figure 3 gives an illustration of the rejection sampling technique.

Though rejection sampling is a very simple method, there are a couple of issues that we encounter. It is not easy to sample a fixed set of points without wasting samples. For shapes with elongated parts, or for shapes with tubular structures, the accept-reject method wastes a lot of samples. The number of samples that get wasted is proportional to the ratio in areas between the bounding rectangle and the object. This ratio can be quite high for objects with parts spread over a large region, such as horseshoes, octopi, or insects. Even for simple shapes such as the circle in Figure 3, a large number of points shown as crosses ('x') are wasted. Our method, which is described below, does not waste any samples, and is therefore able to maintain a constant complexity for all types of shapes.

We make use of the following two steps to overcome the above-stated problems. Firstly, our sampling area is restricted to lie completely within the object's boundary. Secondly, to ensure quick and easy sampling, we ensure that the area we are sampling from is a simple shape. Below, we explain in detail how we propose to sample a fixed number of points without wasting any samples. We call these points as *Dense Points* as they are a dense representation of the shape.

Given a shape S , we first extract its boundary, \mathcal{B} . Then, a set of uniformly spaced points are sampled from the boundary of the object, which we denote as $\mathcal{B}_P = \{\mathcal{B}_{p_1}, \mathcal{B}_{p_2}, \dots, \mathcal{B}_{p_{|\mathcal{B}_P|}}\}$. The boundary constraints makes sure that a point \mathcal{B}_{p_i} neighbours just two other points $\mathcal{B}_{p_{i-1}}$ and $\mathcal{B}_{p_{i+1}}$. We make use of this neighbourhood constraint and perform a Constrained Delaunay Triangulation (CDT) of these points. A CDT ensures that the edges specified as the constraints are retained in the triangulation process.¹³ Once the triangulation is performed, the triangles that lie in the concavities and holes of a shape are removed.¹⁴ We have now ensured that all the triangles that are generated by the triangulation process lie totally within the object's boundary. Figure 4c shows the output of the constrained triangulation. For a given set of N_B points on the boundary, such a Constrained Delaunay Triangulation produces $N_B - 2$ triangles, $T = \{\Delta_1, \Delta_2, \dots, \Delta_{N_B-2}\}$.

Now that the triangles all lie completely within the object's boundary, it is straightforward to sample points from within the shape. For any triangle, Δ_i , with vertices A , B , and C , a random point, p , lying inside the triangle, can be generated using

$$p = (1 - \sqrt{r_1})A + \sqrt{r_1}(1 - r_2)B + \sqrt{r_1}r_2C, \quad (1)$$

where $r_1, r_2 \in [0, 1]$ are two random numbers, independent of each other.¹⁵ A , B , and C , contain the x and y coordinates of the three vertices. The set of points lying inside Δ_i is given by $\mathcal{P}^{\Delta_i} = \{p_1^{\Delta_i}, p_2^{\Delta_i}, \dots, p_{|\mathcal{P}^{\Delta_i}|}^{\Delta_i}\}$. In order to generate $|\mathcal{P}^{\Delta_i}|$ points lying inside the triangle Δ_i , all we have to do is generate $|\mathcal{P}^{\Delta_i}|$ pairs of random numbers (r_1, r_2) , from a uniform distribution, and use Equation 1 to calculate the coordinates of the sampled points.

In order to generate a fixed number of uniformly distributed *Dense Points* (N_{DP}), lying within any arbitrary shape (Figure 4d), we generate samples from all the triangles obtained using the CDT, such that the number of points sampled from each triangle is proportional to the ratio of the area of that triangle to the area of the complete shape. The number of points to sample from triangle Δ_i can be calculated as,

$$|\mathcal{P}^{\Delta_i}| = \frac{A_{\Delta_i}}{\sum_{j=1}^{N_B-2} A_{\Delta_j}} N_{DP}. \quad (2)$$

Where, A_{Δ_i} is the area of triangle Δ_i . Therefore,

$$N_{DP} = \sum_{i=1}^{N_B-2} |\mathcal{P}^{\Delta_i}| \quad (3)$$

The above description shows how a fixed number of sampled points can be generated from any kind of arbitrary shape without wasting any samples. Our method overcomes the two problems that were previously listed in this section. Firstly, the triangulation restricts the sampling area to lie within the shape, thus preventing any sampled points from being wasted. Secondly, we sample from a very simple polygon, a triangle, thus making the sampling of uniformly spaced random points quick and easy. Illustration of the complete procedure is given in Figure 4.

These densely sampled points approximate the interior density of a shape. In order to extract this interior information as comparable features, we make use of the Solid Shape Context (SSC).¹ SSC is generated similar to the well-known shape context descriptor.⁵ We propose to sample the feature locations along the convex hull of the shape. At each feature point, we generate a 2-D histogram of distance and angular bins. The histograms are generated by counting the number of dense points that fall in each bin. This allows us to capture the interior properties of the shape. We encourage the interested reader to refer to an extended version of this paper,¹ where we explain the histogram generation process in detail. In Section 4, we present some results of using the SSC, and show major improvements in image retrieval.

4. EXPERIMENTS

We use the MPEG7 CE-Shape-1 Part B dataset for testing our algorithm. The database consists of 1400 images with a wide variety among them. The database is split into 70 classes, with each class containing 20 example images. The database consists of both rigid and non-rigid objects. The objects in the database have varied levels of translations, rotations, scales, articulations, deformations and occlusions. The objects belonging to a particular class are not only similar by the contour properties, but also by their overall visual similarity. The database is considered as a challenging database as there are many instances where the inter-class object similarity is greater than intra-class object similarity.

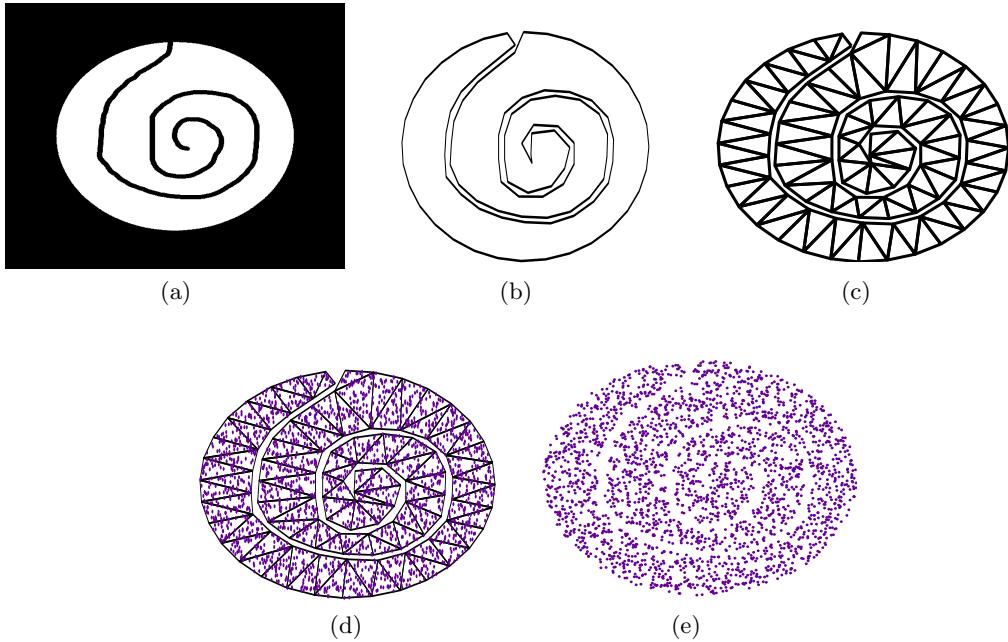


Figure 4: (a) The figure shows the silhouette of a disc with a noisy indent in the contour. (b) Contour of disc extracted from the object shown in (a). (c) Output of the Constrained Delaunay Triangulation. (d) *Dense Points* sampled from inside each triangle according to Equations 1 and 2. (e) Object in (a) represented using the interior points better approximates the object than the contour approximation of the object shown in (b).

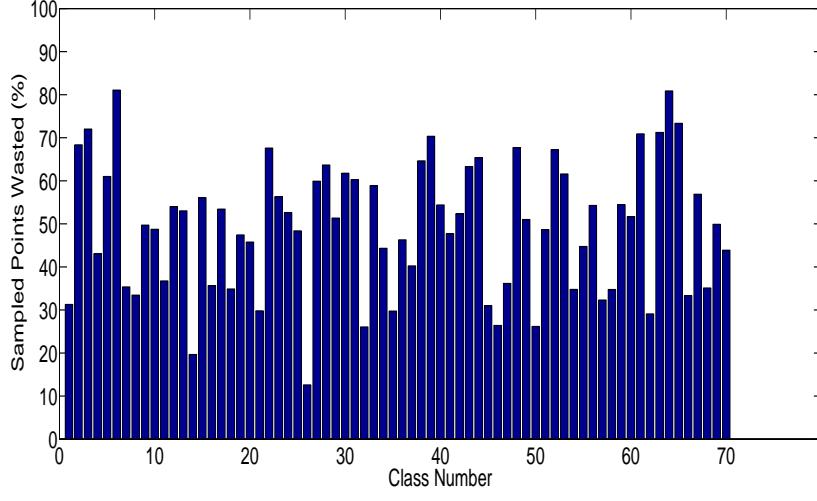
4.1 Accept-Reject v. Constrained Sampling

Before reporting the image retrieval performance using the new descriptor that takes advantage of the dense interior points, we would like to show some statistics about the advantages of using our sampling procedure compared to the simple accept-reject technique. Figure 5 shows a bar chart of the average percentage of samples that were wasted for each of the 70 classes in order to generate 1000 uniformly spaced points from inside each object in the MPEG7 database. As can be seen in the figure, every class requires us to sample more than 1000 points. The wastage is most apparent in classes 6 and 64, which produce the two tallest peaks in the bar chart shown in Figure 5a. Example images from these two classes are shown in Figures 5b and 5c. As mentioned in Section 3, the accept-reject technique will work well only if the shape occupies a considerable area of the bounding box from which the samples are generated. The accept-reject sampling technique required about 2200 points to be sampled in order to generate 1000 points lying inside the shape, on average, across the whole database. This shows an efficiency of less than 50% for the accept-reject sampling procedure. On the other hand, our technique of constrained sampling does not waste a single point as it is guaranteed that all the sampled points lie inside the object's boundary.

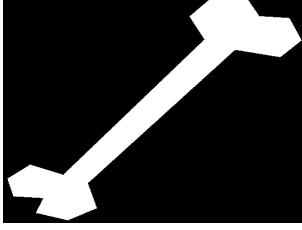
4.2 Image Retrieval

Now that we have verified the advantage of using our sampling technique over the accept-reject method, we proceed to demonstrate the advantage of using dense points in shape retrieval. The performance of the algorithm on the MPEG7 shape database is measured by the Bullseye score. To calculate the Bullseye score, each image is compared to every other image in the database. The top 40 best-matching images are retained, of which at most 20 images can belong to the same class. Of the top 40 best matches, the number of objects belonging to the same class as the template image are counted. This number is divided by 20 to get the Bullseye score for the template image under consideration. The average Bullseye score over all the images in the database gives the Bullseye score for the complete database.

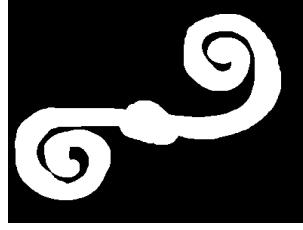
Figure 6 shows the retrieval results for two example objects. The object being matched is shown at the top-left corner of each subfigure. The rest of the objects constitute the top-40 best matches for the object being



(a)



(b)



(c)

Figure 5: (a) The bar chart shows the percentage of samples that got wasted while generating 1000 uniformly spaced points lying inside the object’s boundary, using the simple accept-reject sampling technique. (b) and (c) shows examples from class 6 and class 64, which represent the classes that produce the two tallest bars in the bar chart shown in (a).

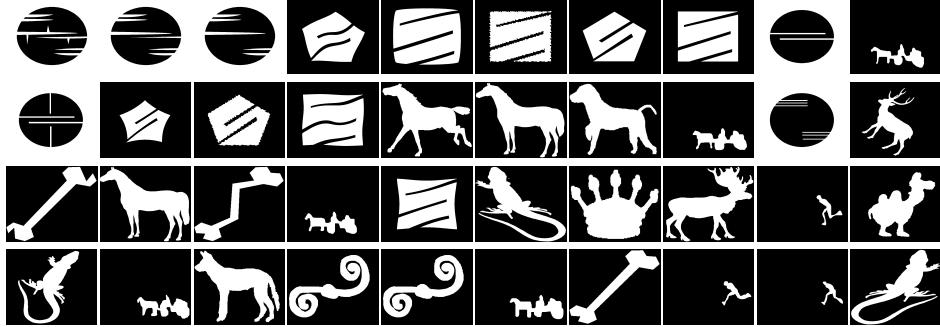
Algorithm	Bullseye Score
IDSC ⁶	85.40%
IDSC+SSC	91.65%

Table 1: Table shows that using SSC significantly helps improve the retrieval results.

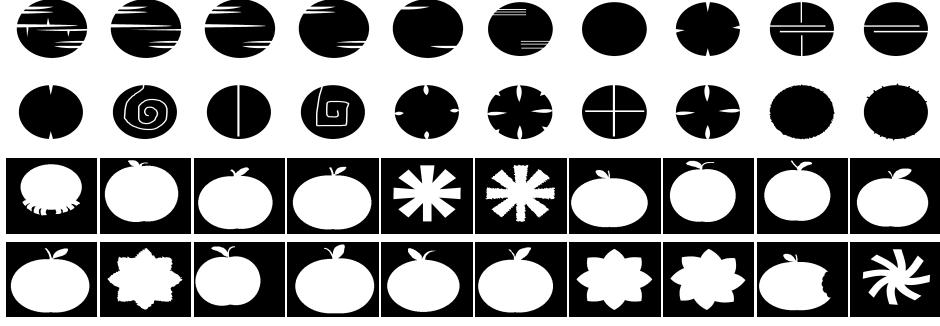
matched. The correct retrievals are the ones where the object is black and the background is white, and the incorrect retrievals are the ones where the object is white and the background is black. We can see that SSC, which uses the dense interior points to compute the shape descriptor, performs much better than IDSC (which is a contour based method) on shapes that have indentations in their contours. Table 1 shows, quantitatively, the improvement that using SSC provides to shape retrieval. We can see that using SSC along with IDSC improves the retrieval results from 85.40% to 91.65%. We encourage the interested reader to refer to the detailed article on Solid Shape Context¹ where details about the experimental settings are explained.

4.3 Extensions to 3-D objects

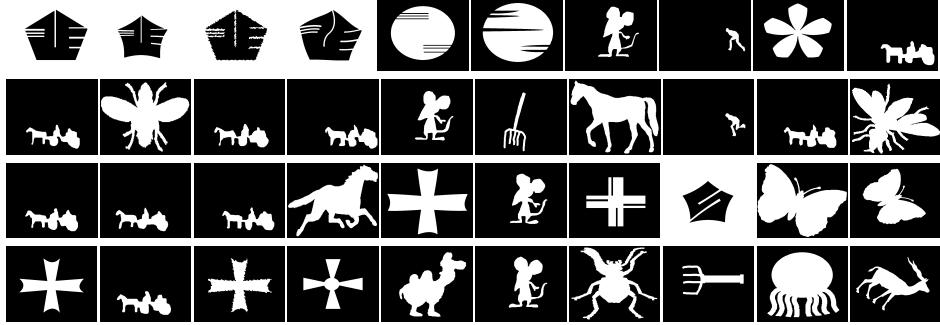
Not only does a dense representation of the objects help in describing 2-D objects, such a representation also helps in describing 3-D objects. Usually, the 3-D objects are represented using a triangulated mesh structure. This, however, does not guarantee that the vertices of the triangulated mesh are uniformly distributed across the surface of the mesh. In such cases of non-uniform distribution of vertices on the 3-D mesh, computing even simple metrics such as the principal components gets affected. We know that if an object is symmetric, then the plane of symmetry is along one of the principal planes of the object. So, in applications that rely on the



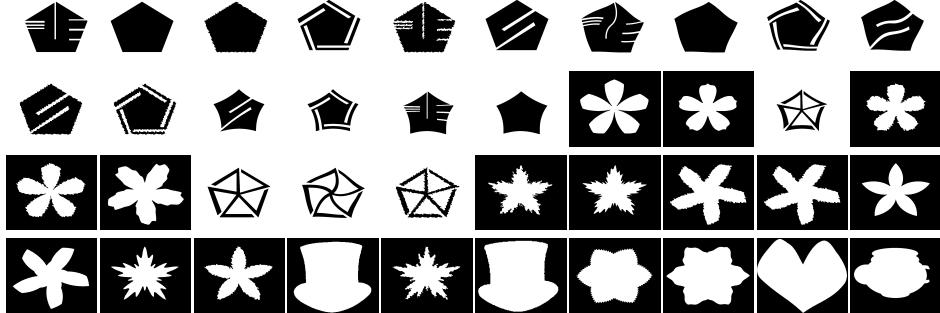
(a) IDSC



(b) SSC



(c) IDSC



(d) SSC

Figure 6: Figure shows the retrieval results for two example objects. (a) and (c) show the results obtained from IDSC, while (b) and (d) show the results obtained from SSC. The object being matched is at the top-left corner of each subfigure. The correct matches are the ones where the object is black and the background is white, and the incorrect matches are the ones where the object is white and the background is black. We can see that SSC performs much better than IDSC for objects with strong indentations in their contours.

identification of symmetry, it is important that one be able to calculate the principal planes accurately. We demonstrate the usefulness of dense points in obtaining accurate symmetry planes. Given a principal plane, ρ , we calculate the symmetry error measure of the object as,

$$\Psi(\rho) = \frac{1}{|P_{left}|} \sum_{i=1}^{|P_{left}|} \min_{j \in \{1, 2, \dots, |P_{right}|\}} \|R_\rho(P_{left}^i) - P_{right}^j\| \quad (4)$$

Here, $\Psi(\rho)$ is the symmetry error measure of the object across the plane, ρ , P is the set of points from the object, $|P_{left}|$ is the number of points to the left of the plane, ρ , $|P_{right}|$ is the number of points to the right of the plane, ρ , P_{left}^i is the i th point on the left side of the plane, and $R_\rho(P_{left}^i)$ is the reflection of the point P_{left}^i across the plane, ρ . We calculate Ψ for all three principal planes and retain the one with the least error measure. We obtain the symmetry error measure by computing the principal planes of the object using just the vertices of the object. We then repeat this procedure by calculating the symmetry error measure using the principal planes that were computed using the densely sampled points from the object.

In Figure 7, we show a 3-D object, a potted plant, taken from the Princeton Shape Benchmark database.¹⁶ The dominant principal component of the object is the axis going from the top to the bottom of the plotted plant. Also, the object is fairly symmetric across the plane containing the principal axis. The image on the left of Figure 7a is the triangular mesh, and on the right is the point cloud obtained from the dense sampling of the mesh. Figure 7b shows the symmetric plane computed for the two input types. We can see that the dense sampling of the points from the mesh helps in producing the correct symmetric plane (right of Figure 7b), while the non-uniform vertices of the triangular mesh produces an incorrect symmetric plane (left of Figure 7b).

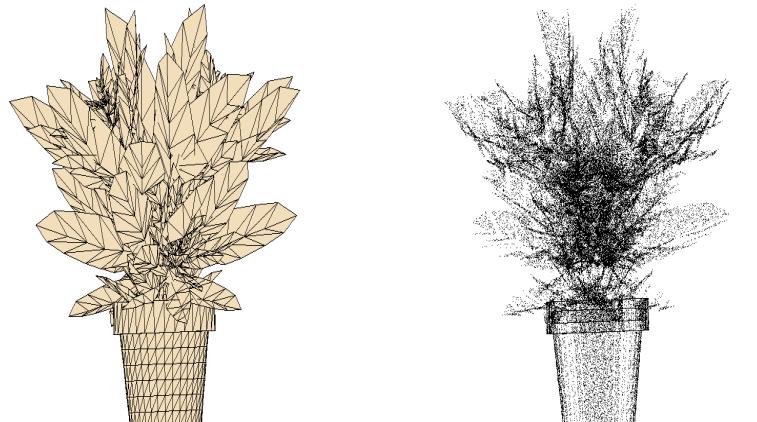
We experimented over all the shapes in the PSB database and found that the dense sampling of the points helps in calculating better symmetry planes for objects whose vertices were not spread uniformly across the surface mesh. For objects whose vertices were uniformly spread across the surface, the improvement that dense points produced, was minimal. In some cases, the error measure obtained from the point cloud was greater than the error measure obtained from using the vertices alone. However, the difference in the error measure was not large. We believe this happens because the dense points are sampled using random number generators, and that the randomization does not always lead to perfectly symmetric sampling of points across the plane of symmetry.

5. CONCLUSION

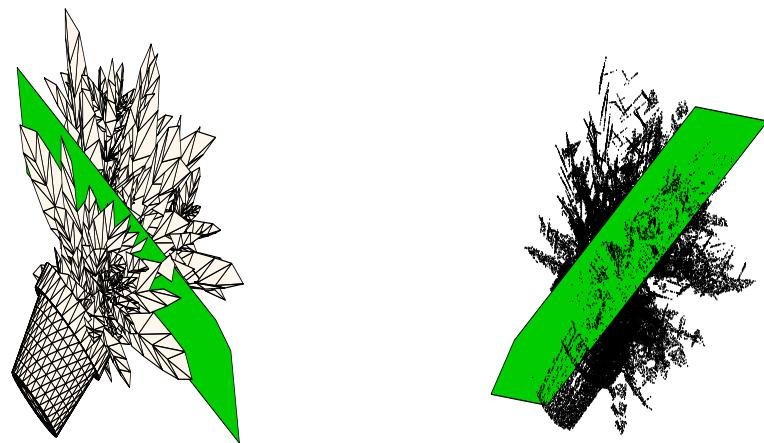
In this paper, we have identified a problem with current day shape matching techniques and proposed that using the shape interior can help in overcoming these problems. We point out that the shapes of the objects are not only recognised by their contour properties, but also by their overall visual similarity. This overall visual similarity can be extracted using interior shape density. We show a simple and effective way of modelling the interior density of the shape. We then use the dense sampled points to generate a variant of the well-known shape context shape descriptor, namely Solid Shape Context (SSC). We show from our experiments that using SSC for shape matching helps gain invariance to indentations in contours. Also, using the dense points helps in extracting more robust shape statistics, such as the principal components of the shape. As a part of the future work, we would like to experiment how this method can be extended to be useful while dealing with voxel data.

REFERENCES

- [1] Premachandran, V. and Kakarala, R., “Perceptually motivated shape context which uses shape interiors,” *accepted for publication at Pattern Recognition* (2013).
- [2] Desolneux, A., Moisan, L., and Morel, J., *[From gestalt theory to image analysis: a probabilistic approach]*, Springer Verlag (2007).
- [3] Sebastian, T., Klein, P., and Kimia, B., “On aligning curves,” *IEEE Transactions on Pattern Analysis and Machine Intelligence* **25**, 116–125 (2003).
- [4] Felzenszwalb, P. and Schwartz, J., “Hierarchical matching of deformable shapes,” in *[Computer Vision and Pattern Recognition]*, 1–8 (2007).



(a)



(b)

Figure 7: (a) A symmetric and elongated object, a potted-plant, represented using a triangulated mesh, and as a point cloud. The point cloud was obtained using the dense sampling procedure. (b) The symmetric plane calculated for the two inputs, is overlayed on them. We can see that using a dense point cloud helps in properly detecting the symmetric plane.

- [5] Belongie, S., Malik, J., and Puzicha, J., “Shape matching and object recognition using shape contexts,” *IEEE Transactions on Pattern Analysis and Machine Intelligence* **24**, 509–522 (2002).
- [6] Ling, H. and Jacobs, D., “Shape classification using the inner-distance,” *IEEE Transactions on Pattern Analysis and Machine Intelligence* **29**, 286–299 (2007).
- [7] Gopalan, R., Turaga, P., and Chellappa, R., “Articulation-invariant representation of non-planar shapes,” in [*European Conference on Computer Vision*], 286–299, Springer (2010).
- [8] Siddiqi, K., Shokoufandeh, A., Dickinson, S., and Zucker, S., “Shock graphs and shape matching,” *International Journal of Computer Vision* **35**(1), 13–32 (1999).
- [9] Sebastian, T., Klein, P., and Kimia, B., “Recognition of shapes by editing their shock graphs,” *IEEE Transactions on Pattern Analysis and Machine Intelligence* **26**(5), 550–571 (2004).
- [10] Xie, J., Heng, P., and Shah, M., “Shape matching and modeling using skeletal context,” *Pattern Recognition* **41**(5), 1756–1767 (2008).
- [11] Temlyakov, A., Munsell, B., Waggoner, J., and Wang, S., “Two perceptually motivated strategies for shape classification,” in [*Computer Vision and Pattern Recognition*], 2289–2296 (2010).
- [12] Hu, R.-X., Jia, W., Zhao, Y., and Gui, J., “Perceptually motivated morphological strategies for shape retrieval,” *Pattern Recognition* **45**(9), 3222–3230 (2012).
- [13] Paul Chew, L., “Constrained delaunay triangulations,” *Algorithmica* **4**(1), 97–108 (1989).
- [14] Shewchuk, J., “Triangle: Engineering a 2d quality mesh generator and delaunay triangulator,” *Applied Computational Geometry Towards Geometric Engineering* , 203–222 (1996).
- [15] Osada, R., Funkhouser, T., Chazelle, B., and Dobkin, D., “Shape distributions,” *ACM Transactions on Graphics (TOG)* **21**(4), 807–832 (2002).
- [16] Shilane, P., Min, P., Kazhdan, M., and Funkhouser, T., “The Princeton shape benchmark,” in [*Shape Modeling International*], (June 2004).