Skeleton Features Distribution for 3D Object Retrieval

Tomoki Hayashi[1], Benjamin Raynal[2], Vincent Nozick[2], and Hideo Saito[1]

[1]Graduate School of Science and Technology, Keio University, Japan
[2]Gaspard Monge Institute, UMR 8049, Paris-Est Marne-la-Vallée University, France

Abstract

In this paper, we propose a 3D shape similarity measurement method using the surface skeleton of a voxelized 3D shape. A set of features are extracted from the skeleton by combining the skeleton and the distance map of the subject. The distribution of these extracted features is used to represent the 3D shape. Two 3D shapes can be compared with a similarity score computed from the distance between their respective features distribution. Our method is robust to noise and to partial occlusion. Moreover, it is scale and rotation invariant. We tested and validated our method with various objects with different shape and topology.

1 Introduction

Content based 3D object retrieval is an important research topic in computer vision and for multimedia community. It results from the improvement of the 3D objects scanning techniques in real scenes. 3D object retrieval has many applications such as computer-aided design, molecular modeling, video games, etc. As a result, numerous 3D objects are accumulated in many databases and it is necessary to use specific methods to retrieve them efficiently. Naturally, these retrieval methods should represent the objects accurately and concisely to compare them efficiently and quickly, independently to any rigid transformation between the compared objects or to noise due to the scanning errors.

In this paper, we present a new retrieval method for 3D shapes represented by voxels, based on the distribution of the features extracted from their surface skeleton. Figure 1 summarizes our retrieval method. We first extract a surface skeleton from an input 3D volumetric object. This surface skeleton is robust to noise and to rigid transformations. We use the geometric relations between the skeleton elements to generate a 2D histogram. This histogram is a compact representation of the considered object. We propose a similarity measure to compare two histograms and express the similarity between the two objects.

This paper is organized as follows: we first introduce the related work about skeleton based retrieval methods and distribution based methods in Sec. 2. Then, Sec. 3 details the skeleton feature distribution used for the object representation. Sec. 4 presents the similarity measure between histograms. Finally, we show some experimental results in Sec. 5, demonstrating the robustness of our method in regard to rigid transformations, noise and missing parts in objects.

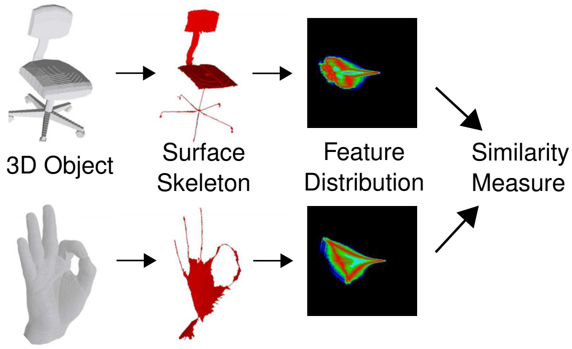

Figure 1. Overview of our retrieval method. First, a surface skeleton is computed from the input objects. Second, a skeleton feature distribution histogram is extracted from the 2D skeletons. Finally, a similarity is measured between the two histograms.

2 Related works

Skeleton is a powerful shape descriptor defined by Blum [1] in 1962, based on a " grass fire" analogy. Imagine a shape as a field covered by dry grass, if you set on fire the contour of the field, then the meeting points of the flame fronts would constitute the skeleton of the shape. In the continuous framework, this definition is equivalent to saying that the skeleton is the set of points which are centers of maximal balls, i.e. balls included in the object and not strictly included in any other such ball. In 3D, we can define two kind of skeletons: curvilinear skeleton and surface skeleton (see Fig. 2 for some illustrations). In this paper, we only consider skeleton based methods for shape retrieval.

Numerous methods based on skeleton have been proposed in order to measure the similarity between 2D shapes. The most famous is probably the shock graph approach, proposed by Siddiqi and Kimia [2]. It consists in the representation by a graph of the variations of maximal ball radius along the skeleton branches and in the matching of such graphs. This method has been improved by Torsello and Hancock [3], by taking into consideration the importance of each branch of the skeleton for the representation of the shape.

Di Ruberto proposed in [4] a direct graph representation of the skeleton, by representing all intersections and ending points by vertices and all the branches by edges. The similarity measure between the original shapes is obtained by measuring the distance between the graphs.

Some other methods use the skeleton without graph representation: a method proposed by Goh [5] consists

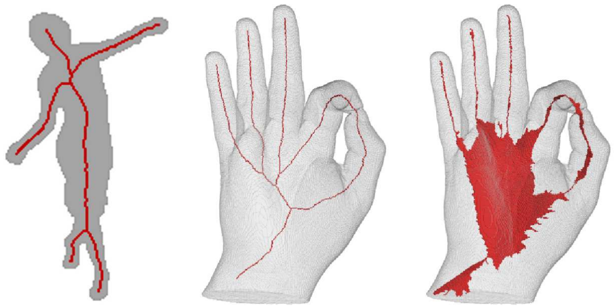

Figure 2. Illustration of the different skeletons. From the left to the right: a curvilinear skeleton of a 2D shape, a curvilinear skeleton and a surface skeleton of a 3D shape.

in segmenting the skeleton in elementary parts that are matched together. Bai and Latecki [6] present a similarity measure based on the skeleton paths, i.e. the curves belonging to the skeleton and linking two ending points.

However, all the methods mentioned above are not compatible with 3D shapes because they require a curvilinear skeleton. In 3D, a curvilinear skeleton does not represent efficiently all the geometry of the shape, contrary to a surface skeleton. Sundar et al. [7] proposed an approach based on the 3D shapes curvilinear skeleton. This method is very efficient for articulated and tubular shaped objects but does not take into consideration the whole geometry of the object.

For our best knowledge, there is no method using surface skeleton information for 3D shape similarity measurement.

Osada et al. [8] defined in 2002 a new approach to the problem of similarity measure between shapes: instead of comparing the features of the shape, the authors proposed to compare the distribution of these features by counting the number of occurrences of each feature. Then the comparison is performed by measuring the similarity between these distributions.

In this paper, we apply this approach to compare the surface skeletons of two shapes by considering the distribution of their features.

3 Skeleton Features Distribution

In this section, we describe how to compute the skeleton features distribution of an object $O \subset \mathbb{Z}^3$. A skeleton feature represents the geometric relationships between pair of voxels included in the surface skeleton. Once all skeleton features are obtained from every pair of voxels, we represent their distribution in a two dimensional histogram. We call this histogram a skeleton features distribution.

3.1 Surface Skeleton Computation

The surface skeleton is obtained using the 6-directional thinning algorithm constrained by 2D-isthmuses (denoted *D6I2D*) defined by Raynal and Couprie in [9]. This algorithm is interesting in our case for three reasons:

- the resulting surface skeleton preserves only the main features of the object and a reconstruction from this skeleton will remove the small perturbations of the object.
- the resulting surface skeleton contains a very low amount of spurious branches, i.e. parts of the skeleton useful for the reconstruction and which are often very sensitive to rotations.
- the algorithm is very fast and hence perform a fast computation of the descriptors.

The main problem resulting of a surface skeleton is its size: if the original object is too small, the skeleton can be inaccurate. On the other hand, if the object is too big, the number of voxels contained in the skeleton can be very large, resulting in a time consuming features computation. In order to solve these problems, the considered objects are resized before the thinning, such its maximal dimension is equals to a fixed value N.

We denote by $S(O)$ the surface skeleton obtained from the resized version of the object O.

3.2 Skeleton Feature

After generating the surface skeleton of the object, we compute the skeleton features represented by the geometric relationship between skeleton voxel pairs. From any pair of voxels $(p, q) \in S(O) \times S(O)$ we can obtain three distances:

- the Euclidian distance between p and q, denoted by $d_{p,q}$
- the radius of the maximal balls for the voxel p and the voxel q, denoted r_p and r_q respectively.

Figure 3 illustrates these distances. By definition, these distances are scale dependent. However, if we consider a ratio of two of them, we obtain a scale invariant measure. Thus, we can represent these three distances by a couple of scale invariant values:

$$SF(p, q) = \left(\frac{r_p}{d_{p,q}}, \frac{r_p}{r_q} \right) \tag{1}$$

Notice that any other ratio of the three distances $d(p, q)$, r_p and r_q can be obtained from the two ratios considered in Eq. 1. Moreover, the constraints $d_{p,q} > 0$ and $r_q > 0$ are always satified since $d_{p,q}$ is measured between different voxels and r_q is measured from p to a voxel out of the object.

3.3 Skeleton Features Distribution

The previous section presents how to compute a feature from a 3D shape and its skeleton. This technique is applied on all the voxel pairs belonging to the skeleton, resulting in the set of all the possible features of the object surface skeleton $S(O)$. This set of skeleton feature is denoted by $\mathcal{SF}(O)$. Let $|S(O)|$ be the number of voxels in $S(O)$, the number of features in $\mathcal{SF}(O)$ is $|\mathcal{SF}(O)| = |S(O)|^2$. Usually, an efficient skeleton contains at least five thousands voxels, hence the number of features is too high for being used directly.

Instead of directly using the features, we propose to use their distribution. This distribution is a bi-dimensional histogram, since each feature is a couple

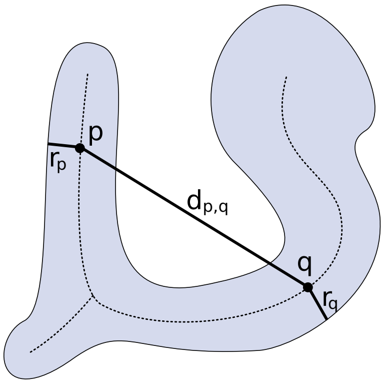

Figure 3. A 2D example of geometric distances for a skeleton feature. The gray area denotes the region of the object. The dash line represents the skeleton. The three measures d, r_p and r_q related to the two points p and q are used for the computation of the feature.

of values. Notice that the two values of the features are ratios, resulting in a dense distribution between 0 and 1 and more and more sparse for high values. In order to obtain an efficient representation of the distribution, we use a logarithmic scale for the histogram, resulting in numerous bins between 0 and 1 and only few bins for largest values.

In addition, as the number of voxels in a skeleton is different for each object, the histogram is normalized such that its integral is equal to 1. We call this normalized histogram the *skeleton features distribution* (SFD) and we denote it by $SFD(O)$. Some examples of SFD are shown in Fig. 4.

4 Similarity measure

In order to compare two objects, we propose to use a similarity measure between their skeleton features distributions. All the SFD have the same dimensions $W \times H$. For a given SFD A, the value of the bin at coordinates (i, j) is denoted by $A_{i,j}$. The sum of all bins values of A is denoted by $|A|$.

Let A and B be two SFD, we consider the *union* of A and B, denoted by $(A \cup B)$, such for any $(i, j) \in \mathbb{N}^2$, $i \leq W$, $j \leq H$:

$$(A \cup B)_{i,j} = max(A_{i,j}, B_{i,j}) \quad (2)$$

In the same way, the *intersection* of A and B, denoted by $(A \cap B)$, is defined such for any (i, j):

$$(A \cap B)_{i,j} = min(A_{i,j}, B_{i,j}) \quad (3)$$

Using Eq. 2 and Eq. 3, we can now define our similarity measure, denoted by $Sim(A, B)$, by:

$$Sim(A, B) = 1 - \frac{|(A \cap B)|}{|(A \cup B)|} \quad (4)$$

This similarity measure ranges between 0 and 1. If A and B are very similar, their intersection will be very similar to their union, resulting in a similarity measure close to 0. If A and B are very different, their intersection will have very small values in comparison to their union, resulting in a similarity measure close to 1.

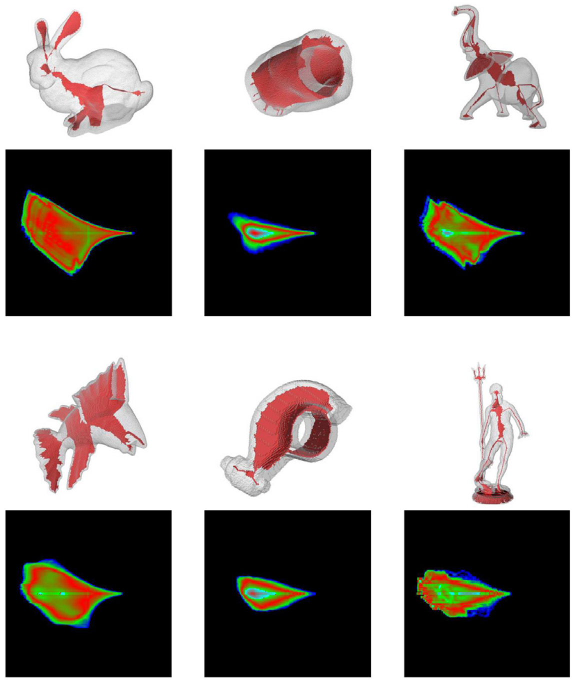

Figure 4. Examples of surface skeletons and skeleton features distribution for different objects. The object is represented by semi transparent voxels, the surface skeleton is represented by red voxels and the skeleton features distribution is represented by the color image bellow the object.

5 Experimental results

All the experimental results have been done on a computer with a 2.33GHz Quad Core CPU and 3 Gigabytes of RAM. The average computation time of a SFD is around 5 seconds and the comparison of a SFD with those of the full database (187 SFD) is done in less than 50 milliseconds.

In order to highlight the good properties of our method, we propose the results obtained for a data base designed for this purpose. Our database is generated from eleven very different objects. For each object, we generate a set of 16 variations:

rotation: rotations of the object (15 and 45 degrees) around each axis → 6 new variations.

scale: rescaling of the object with factor 0.5 and 2 → 2 new variations.

noise: addition of new surface voxels (10, 100, and 1000 new voxels) → 3 new variations.

partial: removal of parts of the object (more than 5 percents of the volume, see Fig. 6 for some examples) → 5 new variations.

By this way, we obtain a set of 187 objects consisting in 17 variations of 11 objects.

In the following, the results are expressed using similarity matrices: each row and each column of the matrix corresponds to one object of the data base. In this visualization of the matrix, the lightness of each element (i, j) is proportional to

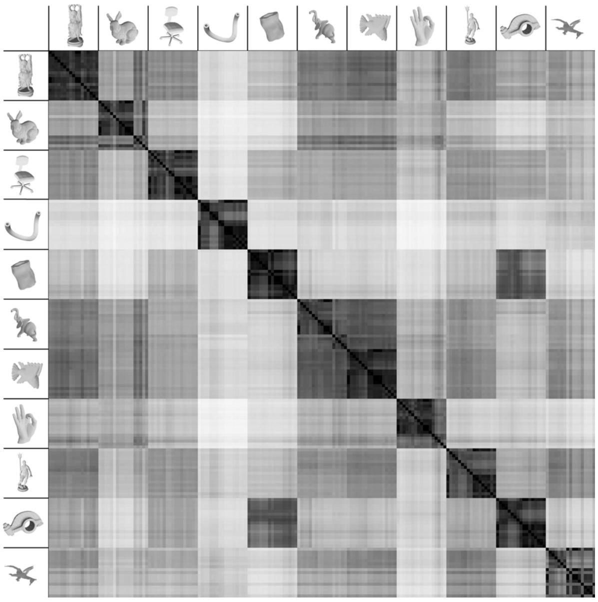

Figure 5. Similarity matrix (187 × 187) of all the variations of the data set.

Figure 6. Examples of variations obtained by parts removal.

$Sim(SFD(O_i), SFD(O_j))$. Darker elements represent better matches whereas lighter elements indicate worse matches. The objects are always sorted in the same order: all the variations are consecutive, starting by the original object, then all its variations in the order of the above description. The similarity matrix of all the data base is shown in Fig. 5.

In order to improve the visibility of the results, we propose to represent only the 17 best matchings for each column, thus the 17 most similar results for each object. In Fig. 7, an element (i, j) is black if and only if the object i is one of the 17 most similar to the object j in our data set. The red squares represent the zones where elements correspond to the similarity between variations of a same object. By this way, we can observe that there is few bad retrievals (black elements out of a red square), especially in the case of original objects, where the average percent of bad retrievals is less than 3.3%.

6 Conclusion

In this paper, we propose a new similarity measure between shapes based on skeleton features distribution. Our method is robust to rigid transformations, noise,

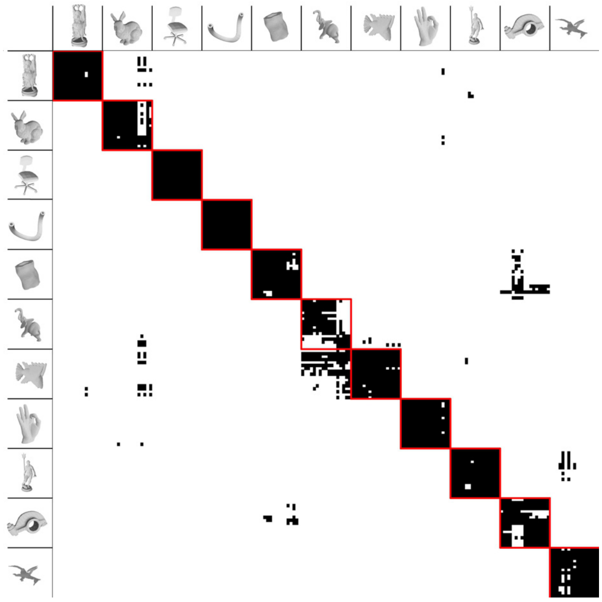

Figure 7. Similarity matrix thresholded in order to keep only the 17 bests results for each column. Red squares represent zones where variations of a same object are compared.

and partial occlusion. In future works, we will try to adapt such approach for a local matching of the different parts of the shape.

References

[1] H. Blum. An associative machine for dealing with the visual field and some of its biological implications. In E.E. Bernard and M.R. Kare, editors, *Biological Prototypes and Synthetic Systems*, volume 1, pages 244–260, NY, 1962. Plenum Press.

[2] K. Siddiqi and B.B. Kimia. A shock grammar for recognition. In *1996 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 1996. Proceedings CVPR'96*, pages 507–513, 1996.

[3] A. Torsello and E.R. Hancock. A skeletal measure of 2D shape similarity. *Computer Vision and Image Understanding*, 95(1):1–29, 2004.

[4] C. Di Ruberto. Recognition of shapes by attributed skeletal graphs. *Pattern Recognition*, 37(1):21–31, 2004.

[5] W.B. Goh. Strategies for shape matching using skeletons. *Computer Vision and Image Understanding*, 110 (3):326–345, 2008.

[6] X. Bai and L.J. Latecki. Path similarity skeleton graph matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(7):1282–1292, 2008.

[7] H. Sundar, D. Silver, N. Gagvani, and S. Dickenson. Skeleton based shape matching and retrieval. *SMI 2003*, 130139, 2003.

[8] R. Osada, T. Funkhouser, B. Chazelle, and D. Dobkin. Shape distributions. *ACM Transactions on Graphics (TOG)*, 21(4):807–832, 2002.

[9] B. Raynal and M. Couprie. Isthmus-Based 6-Directional Parallel Thinning Algorithms. *International Conference on Discrete Geometry for Computer Imagery (DGCI)*, *to appear*.