Feature-Preserving 3D Thumbnail Creation with Voxel-Based Two-Phase Decomposition

Pei-Ying Chiang, May-Chen Kuo, and C.-C. Jay Kuo

Ming Hsieh Department of Electrical Engineering
University of Southern California, Los Angeles, CA 90089-2546
peiyingc@usc.edu

Abstract. We present a feature-preserving 3D thumbnail system for efficient 3D models database browsing. The 3D thumbnail is simplified from the original model so it requires much less hardware resource and transferring time. With topology-preserved 3D thumbnails, the user can browse multiple 3D models at once, and view each model from different angles interactively. To well preserve the topology of the original model, we propose an innovative voxel-based shape decomposition approach, which identifies meaningful parts of a 3D object, the 3D thumbnail is then created by approximating each individual part with fitting primitives. Experimental results demonstrates that the proposed approach can decompose a 3D model well to create a feature-preserving 3D thumbnail.

1 Introduction

The number of 3D models grows rapidly due to the popularity of various applications. Conventional 3D search engines display static 2D thumbnails on the search page for easier browsing. For capturing the best shot of a 3D object, these 2D thumbnails are pre-captured manually, which is very time-consuming. Some researchers attempted to develop systems that can take the best snapshot for a 3D object automatically [1]. This fixed selection rule does not work well for all objects and the result can just as easily capture the wrong features. Yet it is usually impossible to display every important features from a fixed angle.

In this work, we present an innovative feature-preserving 3D thumbnail creation system that helps users browse 3D models efficiently. That is, users can browse multiple 3D models with pre-generated 3D thumbnails, and view each thumbnail from different viewpoints interactively. Usually, to render complicated 3D models simultaneously demands much hardware resource and transfer time that degrades the system performance significantly. The system performance can be improved by rendering the simplified models, i.e. the 3D thumbnails, which requires much less hardware resource and transfer time.

We have developed two approaches to generate a 3D thumbnail; namely, mesh-based and voxel-based approaches. The mesh-based thumbnail creation approach was examined in our previous work [2]. Although many techniques for the mesh-based simplification of 3D models were proposed before, most of them did not address the issue of feature-preserving simplification. For example, the limbs and

G. Bebis et al. (Eds.): ISVC 2010, Part I, LNCS 6453, pp. 108–119, 2010.

the body are important features in a 3D human model, but the existing mesh simplification techniques tend to meld them together. Thus, we proposed a new mesh decomposition technique to preserve the topology of the original model. We first identify significant parts of a 3D model and then simplify each part. The framework in [2] was built upon the surface-based mesh decomposition method [3]. However, the technique in [3] are limited in some scenarios. For example, while an animal mesh can be decomposed into a main body and several protruding limbs successfully, some meaningful parts, such as the head and the neck, cannot be decomposed furthermore. In addition, a decomposed mesh may become fragmented if its adjacent parts were taken apart, which resulted in inaccurate size measurement.

In this work, we develop a voxel-based approach to improve the decomposition result. We convert a 3D polygonal model into a 3D binary voxel grid, then extract its rough skeleton with the thinning operation. The main challenge in the voxel-based approach is that the skeleton obtained using 3D thinning operations or other skeletonization algorithms contains defects that tend to result in a messy 3D thumbnail. To address this problem, we develop the two-phase decomposition process to refine the skeleton. In detail, the skeleton is first decomposed into multiple groups roughly and the voxelized model is decomposed into parts accordingly in the first phase. In the second phase, the skeleton and the voxelized model is re-decomposed more precisely based on the previous result. Finally, the 3D thumbnail will be created by approximating each part with fitting primitives. The significance of this work lies in the two-phases decomposition procedure, where a volumetric model is used in shape decomposition and simplification to overcome several issues associated with the mesh-based approach. The proposed voxel-based approach can decompose a model and extract its skeleton more accurately, and the decomposed model is not fragmented as compared to surface-based approaches. Generally, the resulting 3D thumbnails can well represent the features of the 3D objects.

The rest of this paper is organized as follows. Related works are reviewed in Sec. 2. The framework of the whole system is presented in Sec. 3. The first-phase and the second-phase shape decomposition processes are described in Sec. 4 and 5, respectively. Performance evaluation of obtained results is conducted in Sec. 6. Finally, concluding remarks and future research directions are given in Sec. 7.

2 Review of Previous Work

Various mesh/shape decomposition algorithms were developed to decompose the mesh into small parts based on properties of interest. Shamir [4] provided a thorough survey on mesh decomposition techniques, which can be classified into two major categories depending on how objects are segmented; namely, segmentation is done based on 1) semantic features or 2) geometric primitives.

Methods in the first category is to mimic human perception in psychology or shape recognition to retrieve meaningful volumetric components. Katz and Tal [5] introduced a fuzzy-based cutting method to decompose meshes into meaningful

components, where over-segmentation and jaggy boundaries between sub-objects can be avoided. Further performance improvement can be achieved using multi-dimensional scaling, prominent feature point representation and core extraction. Lin *et al.* [3] proposed another decomposition scheme built upon cognitive psychology, where the protrusion, the boundary strength and the relative size of a part were taken into account. Liu *et al.* [6] developed a part-aware surface metric for encoding part information at a point on a shape. These methods focus on human perception and shape analysis in stead of math formulations. Theories arising from psychology, *e.g.* the minimal rule, separate theory and visual salience are used to analyze crucial components for shape decomposition.

Methods in the second category conduct decomposition based on geometric properties of meshes such as planarity or curvature to create surface patches. Cohen-Steiner *et al.* [7] proposed an error-driven optimization algorithm for geometric approximation of surfaces. They used an idea similar to the Lloyd algorithm and reduced the approximation error by clustering faces into best-fitting regions repeatedly. Other types of proxies were employed to replace mesh patches by Wu and Kobbelt [8] so that their scheme can be more effective for spherical, cylindrical and rolling ball blends. Attene *et al.* [9] proposed a hierarchical face-clustering algorithm for triangle meshes with various fitting primitives in an arbitrary set.

In the practical implementation, we may involve one or more segmentation methods as described above. No matter which category a method belongs to, each method has its own strength and weakness. It is not proper to claim the superiority of an algorithm just because it works well for a certain type of models since it could be deprecated when being applied to others.

3 System Overview

An overview of the proposed thumbnail creation system is depicted in Fig. 1. The main difference between this work and our previous work [2] is that a voxel-based shape decomposition scheme is proposed instead of the mesh-based approach. In this work, the polygonal model is first rasterized into a binary 3D voxel grid and the coarse skeleton is extracted from the volumetric model by the thinning operation, using the tools [10] and [11] respectively. For the rest of the paper, we use *object voxels* to represent the volumetric model and *skeleton voxels* to denote the thinning skeleton as shown in Fig. 2(a). After voxelization and thinning, the first-phase shape decomposition process is used to decompose the skeleton and object voxels. The first phase also provides information to the second phase to adjust the skeleton by eliminating detected defects in the thinning result. The decomposition results can then be fine-tuned in the second phase. The remaining modules include:

1. Parts extraction and pose normalization
   We decompose the original model into several significant parts, and the PCA transformation is applied to each part for pose normalization.

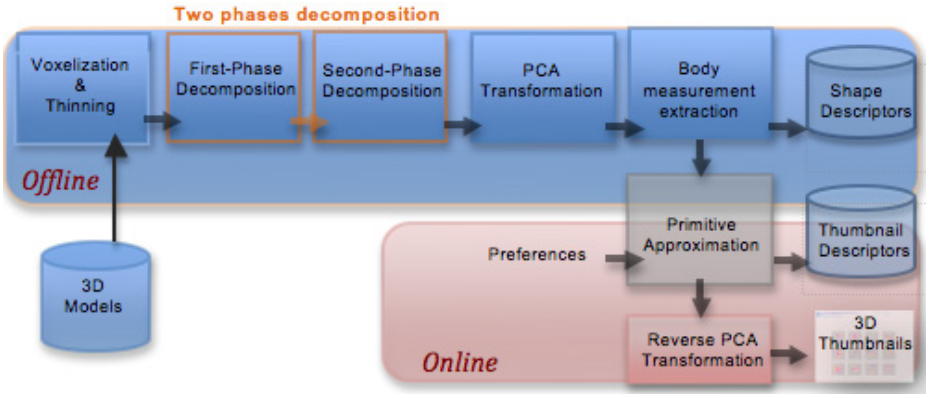

Fig. 1. Overview of the proposed system

2. Body measurement
   The body measurement (*i.e.* the radius of the surrounding surface) of each part is taken along its principal axes.
3. Primitive approximation
   The 3D thumbnail is created by approximating each individual part with fitting primitives and all parts can be put together with the reverse PCA transformation.

The shape descriptor and thumbnail descriptor are generated offline for each model, so the thumbnail can be downloaded and rendered online efficiently. The above steps are built upon [2]. In the following two sections, we will focus on the two-phase decomposition process, which is the main contribution of this work.

4 First-Phase Decomposition

The first-phase decomposition divides the skeleton voxels into multiple groups and then partitions the object voxels accordingly. The decomposition processes of both the skeleton voxels and object voxels are as described below.

4.1 Decomposition of Skeleton Voxels

The skeleton voxels obtained from the thinning operation are a bunch of discrete voxels in the 3D grid. To extract a meaningful skeleton from them to represent a 3D model, we define a rule to link discrete voxels. First, we classify the skeleton voxels into three categories:

- End_{SK}: the voxel which has only one neighbor;
- Joint_{SK}: the voxel which have more than two neighbors;
- Normal_{SK}: the voxel which has exactly two neighbors.

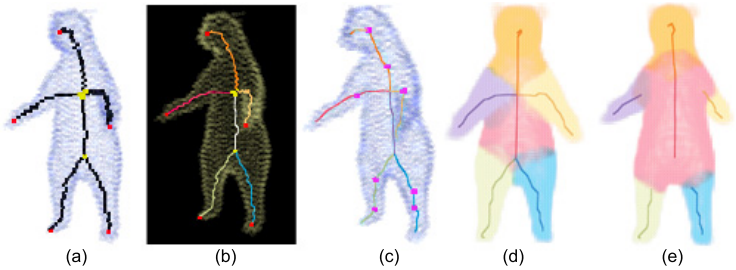

Fig. 2. An example to illustrate skeleton classification and decomposition: (a) object voxels representing a volumetric model are shown in light blue, and skeleton voxels classified as End_{SK}, Joint_{SK}, and Normal_{SK} are shown in red, yellow, and black, respectively; (b) skeleton voxels are decomposed into multiple groups and shown in different colors; (c) turning points (Peak_{SK}) representing local peaks are shown in purple dot; (d) the first-phase shape decomposition result; (e) the ideal shape decomposition for the base part is shown in red

The classification task is easy. That is, we can simply check the 26-adjacent voxels in a $3 \times 3 \times 3$ grid. Once skeleton voxels are classified, we link adjacent skeleton voxels that belong to the same group. We start to create a group with one of the End_{SK} or Joint_{SK} voxels and link it to the next adjacent voxel until another $\text{End}_{SK}/\text{Joint}_{SK}$ is met. As a result, all skeleton voxels can be linked and divided into different groups such as the example shown in Fig. 2(b).

After all skeleton voxels are linked, we extract the turning point so that the model can be decomposed more precisely. The turning point of a skeleton, denoted by Peak_{SK}, represents the local peak of the skeleton and can be extracted by analyzing the curve of the skeleton. Fig. 2(c) shows an example of extracted turning points. It is worthwhile to point out that we do not partition the skeleton with turning points in the first-phase decompostion since this will complicate the following process. The usage of the turning points will be discussed in Sec. 5.2.

4.2 Decomposition of Object Voxels

After the decomposition of skeleton voxels, object voxels which represent the volumetric model can be decomposed accordingly. Although each object voxel could be assigned to its nearest skeleton voxel by computing their Euclidean distance, this do not work well in general. For example, the distance from a finger tip to anther finger tip is not equal to their Euclidean distance since there is no straight path between them. Thus, we define the distance between object voxel V_j to skeleton voxel K_i as

$$\begin{aligned} d(V_j, K_i) &= \|V_j, K_i\| \ \text{, if a straight path } \overline{V_j K_i} \text{ exists} \\ &= \text{infinity , otherwise.} \end{aligned}$$

To check if a straight path exists between V_j and K_i, we create a line segment $\overline{V_j K_i}$. Voxels along this line segment can be derived by interpolation. If there is

an empty voxel lies within $\overline{V_jK_i}$, the distance between them are set to infinity. Accordingly, object voxels can be assigned to different parts according to their belonging skeleton voxels.

The object voxels are classified into two types based on the relationship with their neighbors; namely, *surface* and *interior* voxels. The object voxel that does not have 26 neighbors in a $3 \times 3 \times 3$ grid is called a surface voxel. Otherwise, it is an interior voxel. Since our objective is to estimate the radius of the surrounding surface along the skeleton, our interest lies in the decomposition of surface voxels. Furthermore, we would like to reduce the computational complexity. Based on the above two reasons, we focus only on the decomposition of surface voxels and ignore interior voxels. Finally, an example of the first-phase decomposition result is shown in Fig. 2(d), where each surface voxel is assigned to its nearest skeleton voxel to form a group of the same color.

After the first-phase decomposition, each skeleton voxel, K_i, has a associated list, $List(K_i)$, recording its associated surface voxels. This list is used to calculate the average radius, $Radius(K_i)$, of its surrounding surface. The information of the associated surface voxels and the average radius will be used in the second-phase decomposition.

5 Second-Phase Decomposition

The skeleton of a muti-tubular objects, such as animals, normally has its protruding skeletons intersecting with the base skeleton. There are two kinds of mis-decomposition in the first-phase: Object voxels which belong to the base part are mistakenly assign to adjacent protruding parts, and vice versa. Comparing Fig. 2(d) to the ideal decomposition in Fig. 2(e), a portion of the main body is mistakenly classified to the arm while the main body is mistakenly divided at the middle. The protruding skeleton that cuts across the boundary makes the surrounding area ambiguous and difficult to decompose. To address the problem, the base part and the protruding parts of a 3D model will be identified in the second-phase decomposition. The decomposed skeleton voxels are re-adjusted by invalidating the redundant segment and extending the base skeleton which was mistakenly divided at the intersection. Afterwards, the object voxels will be re-decomposed accordingly.

5.1 Identification of Base Part

To identify the base part, we define weighted accumulated distance for each part P_i as

$$\rho(P_i) = d_E(cen(P_i), cen(P)) \times s(P_i) + \Sigma_{\forall j \neq i}(d_P(mid(P_i), mid(P_j)) \times s(P_j)), \quad (1)$$

where $cen(P_i)$ and $cen(P)$ are the center of mass of P_i and the model P, respectively, $d_E()$ is the Euclidean distance, $s(P_i)$ is the number of object voxels belonging to P_i, and $mid(P_i)$ is the skeleton voxel of P_i that is closest to $cen(P_i)$.

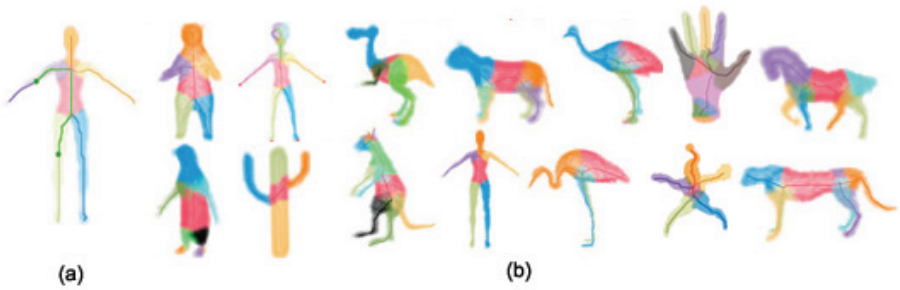

Fig. 3. The base part identification procedure: (a) the path along the skeleton from one part to another, where the green dots represent centers of different parts; (b) identified base part shown in red

Since the distance between two parts is not equal to their Euclidean distance, we define the the distance from P_i to P_j as

$$d_P(mid(P_i), mid(P_j)) = \|path(mid(P_i), mid(P_j))\|, \tag{2}$$

where $path(mid(P_i), mid(P_j))$ contains all the skeleton voxels along the path from the $mid(P_i)$ to $mid(P_j)$, and $\|.\|$ is the path length. The idea is illustrated in Fig 3(a).

Thus, the base part which is closer to all other parts can be identified by finding the part with the minimal accumulated distance. The experimental results of this algorithm are shown in Fig. 3(b).

5.2 Adjustment of Decomposed Skeleton Voxels

To adjust the skeleton, we first constrain the segment of the protruding skeleton to its own region. Second, we extend the base skeleton by merging the protruding skeleton selectively. Finally, the protruding part is further fine-tuned by the usage of turning points. An example is illustrated in Fig. 4.

Adjusting Protruding Skeleton. For a protruding skeleton whose orientation is different from the base skeleton (*i.e. the angle between their orientations is larger than a threshold*), we aim to find its boundary and invalidate its segment that goes beyond the boundary. The boundary can be detected using the property that the surrounding surface radius along the protruding skeleton toward to the base part increases dramatically such as the surrounding surface along the arm as shown in Fig. 2(c). The corresponding skeleton voxel is called the cutting point. Then, skeleton voxels from the cutting point to the intersection with the base part should belong to the base part rather than the protruding part.

In practice, the surrounding surface radius can vary and it is difficult to define a normal range of the surface radius throughout the entire protruding part. To address this problem, we narrow down the search region for the boundary by

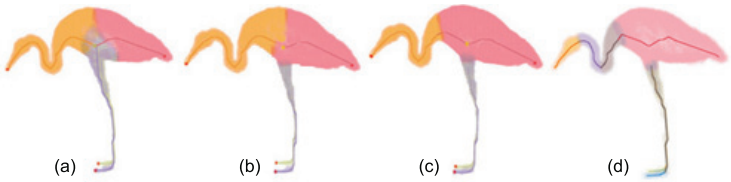

Fig. 4. The adjustment of decomposed skeleton voxels: (a) the result from the first-phase decomposition; (b) invalidating the segment of the protruding skeleton that goes beyond its boundary; (c) extending the base skeleton by merging a segment of a protruding skeleton, (d) dividing groups into sub-groups by Peak_{SK}. As shown in the example, the protruding parts such as the legs and the neck of the crane model are decomposed into subgroups with the selected turning point.

considering the surrounding surface radii along the base skeleton. That is, the average radius δ_1 along the base skeleton is the minimum threshold and the largest base radius δ_2 is the maximum threshold of the search region. Now, consider a protruding part that has successive skeleton voxels $\{ SK_1, SK_2, \ldots, SK_n \}$, and SK_n is the end voxel that is connected to the base part. We calculate the Euclidean distance from each skeleton voxel to SK_n. Only the skeleton voxel whose distance to the base part is within the search range $[\delta_1, \delta_2]$ will be examined to see if it is the cutting point. That is, if the successive skeleton voxels $\{ SK_i, \ldots, SK_j \}$ has the distance to the base within $[\delta_1, \delta_2]$, we search for the cutting point among SK_i to SK_j. If no cutting point is found in this region, we cut at SK_j instead.

After the cutting point is selected, all skeleton voxels from the cutting point to the joint of the base skeleton are removed from the protruding part. The surface voxels previously assigned to these invalidated skeletons are re-assigned to one of the closest base skeletons. Fig. 4(b) is an example of improved decomposition result by adjusting protruding skeletons (*i.e.* the skeleton of the leg).

Extending Base Skeleton. For each protruding group whose orientation is similar to the base group, we aim to merge the base skeletons which was partitioned inaccurately, such as the front body in Fig. 4(a). The decision of mergence/disjoint is made based on two criteria. One is to find the protruding boundary, where the surrounding surface radius increases dramatically. The other is to determine the turning point along the skeleton. Even if the radius deviation along the protruding skeleton is small, the base part should not be extended to the position where the protruding part starts to bend, such as the boundary between the crane's neck and the body as shown in Fig. 4(c). As a result, the search region for the cutting point is from the joint to the closet turning point. If there is no cutting point found in the region, we cut at the turning point.

Fine-Tuning of Protruding Parts with Turning Points. As the final step, we divide each protruding group into subgroups according to the curvature of

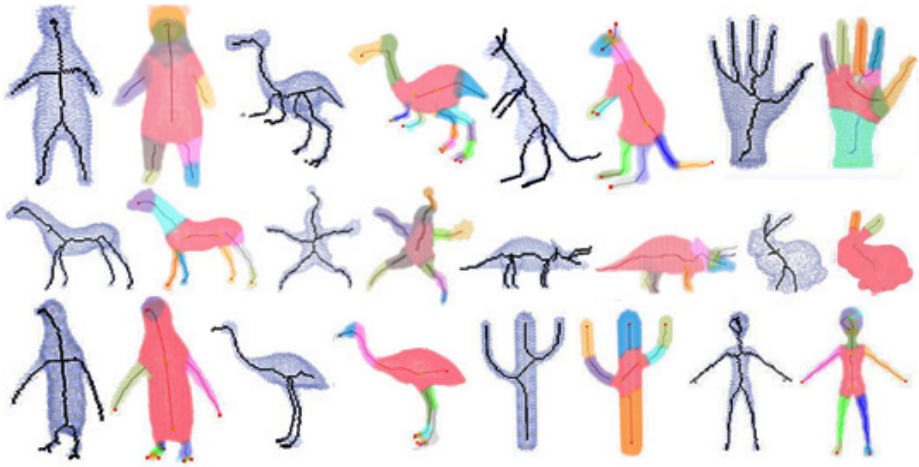

Fig. 5. Comparison of skeletons obtained with the thinning operation and the refined skeleton

its skeleton. If the joint between two subgroups is bended, we can detect the turning point and divide them accordingly. For example, the leg part in Fig. 4(c) will be further divided into a leg and a foot as the result shown in Fig.4(d).

5.3 Re-decomposition of Object Voxels

Finally, a re-decomposition process is carried out for object voxels. Here, we do not need to re-calculate the closest skeleton voxel for each object voxel. Object voxels that belonged to invalidated skeleton voxels in the first phase have already re-assigned to the closest base skeleton in the second phase, and the assignment of all other object voxels is not changed since they still belong to the same closest skeleton voxel. Instead, we only need to assign the new group id to each object voxel according to the skeleton re-decomposition result. Each object voxel is assigned to the same group as its associated skeleton voxel.

6 Experimental Results

In the experiments, we applied the proposed two-phase decomposition approach to a collection of 3D models. All of the models were pre-converted into the same obj format, normalized to the same range, and voxelized to the corresponding volumetric models. In the voxelization process, the smaller size the voxel is chosen (higher resolution), more features can be captured. However, it demands larger memory and its thinning result tends to contain more noise. We decided to use a grid of size $80 \times 80 \times 80$ since it captures the most important features with a reasonable amount of memory while unnecessary details could be reasonably discarded. In addition, the thinning process may take very long time

Fig. 6. Thumbnail results obtained by the proposed voxel-based approach, where the original 3D models are decomposed into multiple parts and each part is approximated by a fitting primitive

to conduct for higher resolution models. For example, it took about 27 mins to finish thinning a bunny model which was voxelized into $500 \times 500 \times 500$ grid. On the other hand, it took less than 1 sec for thinning of the same model which was voxelized into $80 \times 80 \times 80$ grid.

The experiment was run on a desktop computer with an Intel Core 2 Duo 2.53 GHz CPU, and 4G RAM. The first phase decomposition took 0.83 sec and the second phase decomposition took 0.35 sec on the average. Assigning object voxels to different parts in the second phase took much less time than the first phase since we only re-assigned the object voxels whose associated skeleton voxels were invalidated. The average processing time for create a thumbnail descriptors with our voxel-based approach was about 3.6 second.

The two-phase decomposition process refines the skeleton obtained from the thinning algorithm. The protruding skeleton that goes beyond the boundary is removed, the base skeleton is extended and each part is segmented more accurately. Fig. 5 shows the improvement of the skeleton re-decomposition result, both the original skeleton decomposition and the improved skeleton decomposition result are illustrated. All identified base parts are shown in red color and protruding parts are shown in different colors. Each model is decomposed into its significant parts more accurately with the two-phase decomposition procedure. Except for the decomposition process, the other modules described in Sec. 3 remain the same as those in [2]. The significant parts of a model were approximated by fitting primitives individually.

Several created thumbnails, each of which consists of 90 primitives, are shown in Fig. 6. The thumbnail results generated by the proposed voxel-based approach preserve more details of the original models. Some subgroups that could not be separated by the surface-based approach can be separated now. Fig. 7 shows several examples whose significant parts are still well preserved even when the models are greatly simplified.

We compare the surface-based approach in [2] and the voxel-based approach in this work in Fig. 8. Although the surface-based approach can identify the main body and several protruding parts as shown in Fig. 8(a_1). It has two main

Fig. 7. Thumbnails approximated by a different number of primitives by the voxel-based approach, where subparts are not as easily breakable as those obtained by the surface-based approach as presented in [2]

shortcomings. First, the surface-based decomposition is less precise; namely, some meaningful parts which do not directly protrude from a base part cannot be further decomposed. For example, the foot and the head cannot be further separated from the leg and the neck, respectively. Second, the decomposed mesh may not represent the shape well for measurement since it become fragmented while its adjacent parts were taken apart. For example, there are missing pieces in the simplified crane model as shown in Fig. 8(a_2). Its skeleton and the body measurement extracted from the base part leans toward the upper body due to the missing pieces on the bottom as shown in Fig. 8(a_3). These shortcomings are resolved by the proposed voxel-based approach. Figs. 8 (b_1)-(b_3) show the improvement of the skeleton and the body measurement using the voxel-based approach. Besides, the file size of a created thumbnail is similar as we described in [2] since the 3D thumbnail is constructed with the same primitive in both approaches; The size of a thumbnail composed of 90 primitives is about 8KB.

Finally, it is worthwhile to point out one shortcoming of the proposed voxel-based approach. The decomposition result is highly affected by the qualify of the skeleton obtained in the beginning stage. If the skeleton does not represent the correct structure of an 3D model well, the proposed approach fails to decompose the model. Please see the last two thumbnails in Fig. 5. First, the skeleton in the middle part of the cactus model is skewed, and it does not have the same direction as the upper part and the lower part. Thus, the base part of the cactus fails to extend to the right direction. Second, the alien model has two big holes

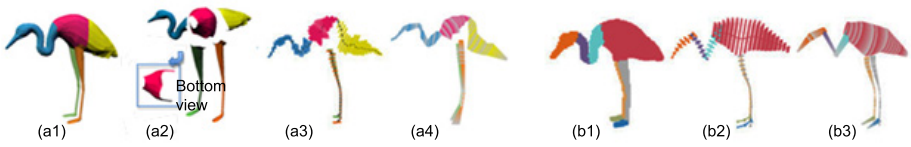

Fig. 8. The comparison of the surface based approach (a) and the voxel-based approach (b). (a_1) The mesh decomposed by the surface-based technique [3]; (a_2) the bottom view of the decomposed base part with several pieces missing; (a_3) the extracted skeleton and the body measurement; (a_4) the resultant thumbnail shows the base part is leaning upward, since the bottom pieces are missing; (b_1) the volumetric shape decomposed by the voxel-based approach; (b_2) the improved skeleton and body measurement; and (b_3) the improved resultant thumbnail.

in the face and the skeleton obtained with thinning cannot capture this feature correctly. As a result, one half of the face is missing in the created thumbnail. To improve these thumbnail results, we need a better skeletonization technique for 3D models, which is a future research item.

7 Conclusion and Future Work

An innovative voxel-based two-phase decomposition approach was presented to resolve limitations encountered in [2]. In this work, the skeleton is decomposed into multiple groups and then the volumetric model is decomposed into significant parts accordingly. The significant parts of a 3D model can be well-preserved by its thumbnail representation even the original model is greatly simplified. As compared with the surface-based approach in [2], the voxel-based approach can represent the shape of each part better and decompose the model more accurately.

There are three possible future extensions of this work. First, in the case that the skeleton of a 3D model extracted by the thinning process fail to represent the structure of the object, the decomposition result can be severely affected. More advanced skeletonization techniques are needed to extract a finer skeleton. Second, different primitive types can be used in primitive approximation for performance comparison. Last, the textures of 3D models can also be considered as possible enhancement.

References

1. Mortara, M., Spagnuolo, M.: Semantics-driven best view of 3d shapes. In: IEEE International Conference on Shape Modelling and Applications 2009, vol. 33, pp. 280–290 (2009)
2. Chiang, P.Y., Kuo, M.C., Silva, T., Rosenberg, M., Kuo, C.C.J.: Feature-preserving 3D thumbnail creation via mesh decomposition and approximation. In: Qiu, G., Lam, K.M., Kiya, H., Xue, X.-Y., Kuo, C.-C.J., Lew, M.S. (eds.) PCM 2010. LNCS, vol. 6298, Springer, Heidelberg (2010)
3. Lin, H.Y.S., Liao, H.Y.M., Lin, J.C.: Visual salience-guided mesh decomposition. IEEE Transactions on Multimedia 9, 45–57 (2007)
4. Shamir, A.: A survey on mesh segmentation techniques. Computer Graphics Forum 27, 1539–1556 (2008)
5. Katz, S., Tal, A.: Hierarchical mesh decomposition using fuzzy clustering and cuts. In: SIGGRAPH 2003, pp. 954–961. ACM, New York (2003)
6. Liu, R.F., Zhang, H., Shamir, A., Cohen-Or, D.: A part-aware surface metric for shape analysis. Computer Graphics Forum (Eurographics 2009) 28 (2009)
7. Cohen-Steiner, D., Alliez, P., Desbrun, M.: Variational shape approximation. In: SIGGRAPH 2004, pp. 905–914. ACM, New York (2004)
8. Wu, J., Kobbelt, L.: Structure recovery via hybrid variational surface approximation. Computer Graphics Forum 24, 277–284(8) (2005)
9. Attene, M., Falcidieno, B., Spagnuolo, M.: Hierarchical mesh segmentation based on fitting primitives. Vis. Comput. 22, 181–193 (2006)
10. Binvox. 3D mesh voxelizer, `http://www.cs.princeton.edu/~min/binvox/`
11. Thinvox. 3D thinning tool, `http://www.cs.princeton.edu/~min/thinvox/`