

Shape Map Method for 3D Body Scanning Information Storage

Peng SIXIANG^{*a}, Chan CHEE-KOOI^a, Ameersing LUXIMON^a, W.H. IP^b

^a Institute of Textiles & Clothing, Hong Kong Polytechnic University, Hong Kong, SAR People's Republic of China;

^b Department of Industrial and Systems Engineering, Hong Kong Polytechnic University, Hong Kong, SAR People's Republic of China

Abstract

Body surface scanning data are mostly stored in point-cloud format currently. This paper describes a novel body surface profile information recording method for the human body, which called shape map format. It adopts novel surface morphological information processing methodology, which is able to retrieve and store both the two-dimensional (2D) and three-dimensional (3D) information of the waist to hip section. Compared to the point-cloud data format, its presentation is more intuitive; its resolution rate is higher; its sampled point distribution is better for surface modeling and its size is much smaller, which is only about 12% of the original form. We have designed two data processing modules in Matlab that convert the body surface scanning data from point-cloud format to shape map format and conversely. Its performance have been evaluated using the 3D scanning data of the waist to hip human body section sampled using a [TC]² 3D scanner.

Keywords: 3D scanning, shape map, information, storage, compress

1. Introduction

3D scanning technique is widely used in various research areas, such as health, apparel fit, clinical nutrition [1-4]. It is also an essential part for 3D body model building, which plays an important role in several domains such as computer-aided design [5], facial expression [6], and 3D digital human body model analysis [7]. In previous research, the 3D scanning equipment digitized/sampled the object's surface point by point and stored in the form of point cloud data [8]. However, a point cloud data of body could occupy up to several Megabyte (MB), it becomes costly when the data base grows large. For example, SizeUSA, which is a comprehensive sizing survey of the U.S. population, conducted by the [TC]² company, scanned nearly 11,000 individuals. The [TC]² 3D scanner output the scanning result in .obj format file, which contains the 3D Cartesian coordinates of the surface points and the surface information which is based on the surface points using 3D modelling technique. The size of the .obj file of a full human body is about 3.5 MB. With only the point-cloud information, its size still can reach about 2 MB per person. So the SizeUSA body scanning point-cloud data set will occupy about 22 Gigabyte (GB), causing a significant load for storing. Another point is that, these data cannot be used directly, as further processing is needed before data retrieving or image processing. The points in the point-cloud data spread unevenly on the surface of the scanned subject, which requires pre-processing such as filtering and hole-filling before surface can be constructed on them. It will be a very time-consuming task when the size of data base is huge.

To solve these problems, we developed a novel body surface profile information recording method for the human body, which is called shape map format. It processes the point-cloud data by constructing new sampling points and storing as portable network graphics (PNG) format files which not only contains all the original points but also increases the scanning data's resolution rate; decrease the data storing size, provides a better support for surface modeling and presents in a more intuitive way. We have designed two data processing modules in Matlab that be able to convert the body surface scanning data from point-cloud format to shape map format and conversely. We tested the shape map method by converting point-cloud data of waist to hip body section of 75 individuals using the [TC]² 3D scanner. The results showed that the shape map data occupied about 12% storage room and improved the resolution about 266%.

2. Methodology

2.1. From point-cloud data to shape map data

The 3D scanning equipment scans the sample object and stores the 3D location information of the points in point-cloud data which contains only their Cartesian coordinate's values. Figure 1 left shows all the points of a point-cloud data in 3D. Firstly the human body is segmented into head, trunk, left hand, right hand, left foot and right foot using body landmark as shown in figure 1 right. All these parts appear in the shape where all the points are on the surface of an irregular cylinder in 3D space (see figure 2).

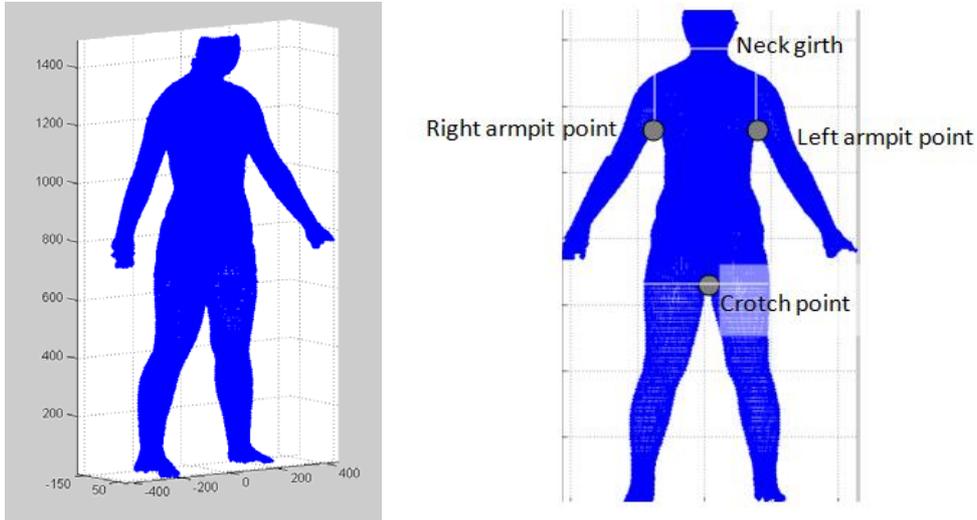


Fig. 1. Left: points in 3D based on point-cloud data information; Right: segment the body with body landmark.

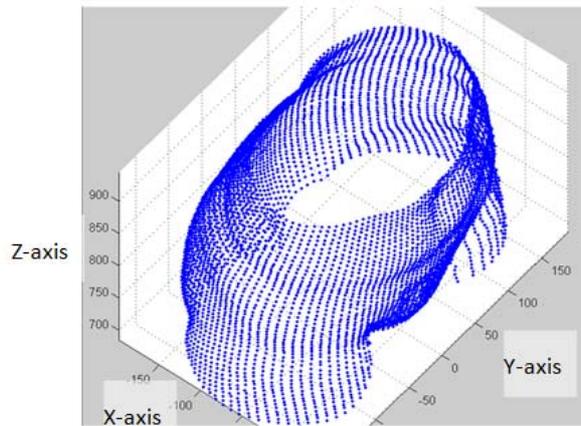


Fig. 2. Waist to hip section of a scanned body surface.

Secondly, all the points will be aligned to where the $Point_{mean}$ is on the Z-axis as figure 3 shown, while $Point_{mean}$ is defined as following:

$$Point_{mean}(x, y) = \overline{Point_{all}(x, y)} \quad (1)$$

Where $Point_{mean}$ is the mean point and $Point_{all}$ is all the points in this section.

So the alignment process can be presented as following:

$$Point_{aligned}(x, y) = Point_{all}(x, y) - Point_{mean}(x, y) \quad (2)$$

Where $Point_{aligned}$ is the points aligned.

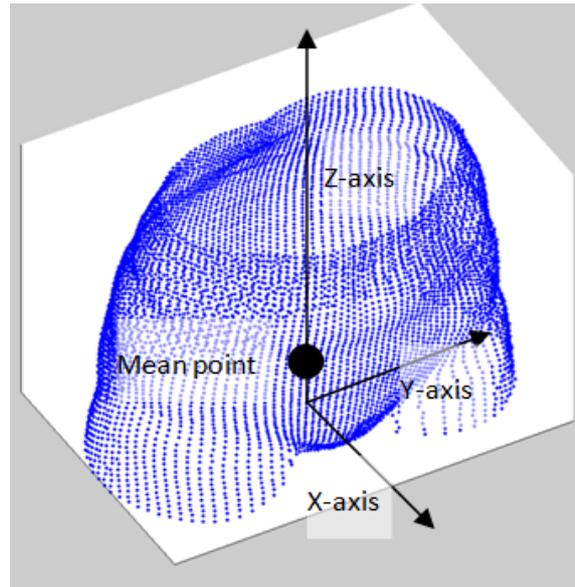


Fig. 3. Waist to hip section of a scanned body surface.

Thirdly, $Point_{aligned}(x,y,z)$ will be converted into Polar coordinate to get $Point_{aligned}(r,\Theta,z)$ (see figure 4 left). It is obvious that the points in point-cloud data are sampled unevenly from the object's surface. However, a uniform distribution of the surface is often required to guarantee a hole-free rendering of the surface [9]. To achieve an even distribution of surface we employ Gridfit method [10]. The Gridfit estimates a surface on a 2d grid, based on scattered data $Point_{aligned}(r,\Theta,z)$. It adopts a modified ridge estimator to generate the surface approximate the $Point_{aligned}(r,\Theta,z)$, where the bias is toward smoothness. It is not an interpolant. Its goal is a smooth surface which is similar to the human surface shown as figure 5. The 201×101 sized evenly distributed grid on surface has 20301 intersection points, and the location of these points, $Point_{grid}(r,\Theta,z)$, are regarded as the sampled point for shape map data.

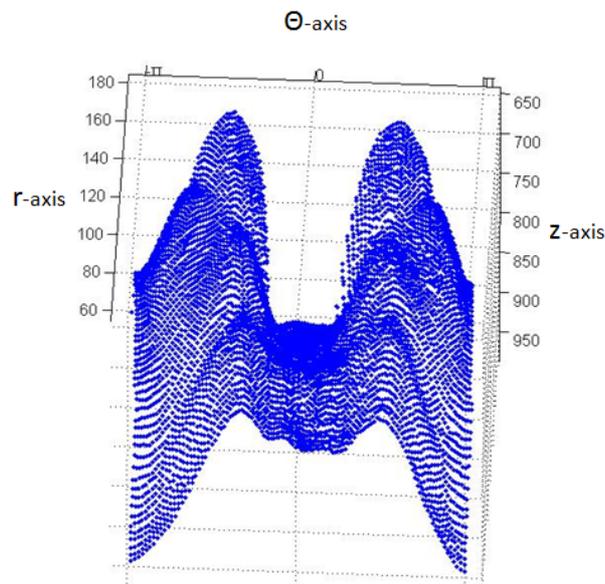


Fig. 4. $Point_{aligned}(r,\Theta,z)$ in 3D space.

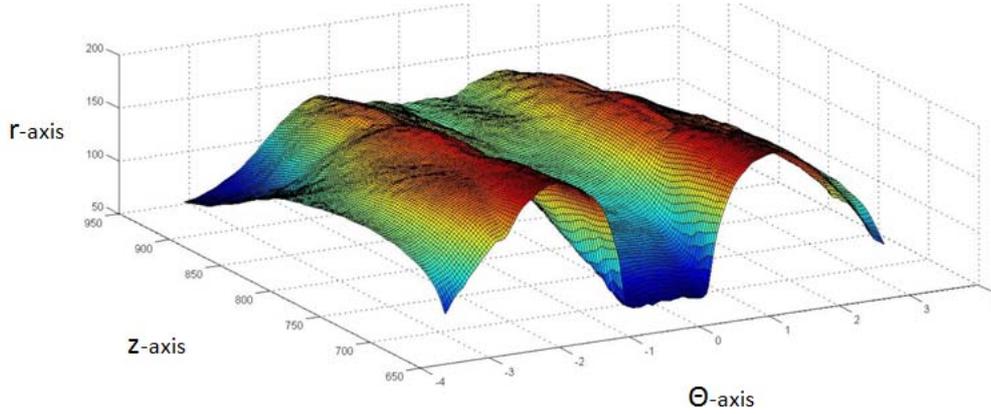


Fig. 5. $Point_{aligned}(r, \Theta, z)$ in 3D space.

Finally, a PNG file is created to store the $Point_{grid}(r, \Theta, z)$ in the form of gray level value. To achieve this, we converted the $Point_{grid}(r, \Theta, z)$ into a 2D matrix $ShapeMap(x', y')$, which is defined as

$$x' = \frac{\Theta + \pi}{Step_x} + 1 \quad (3)$$

$$y' = \frac{z - \min(z)}{Step_y} + 1 \quad (4)$$

$$ShapeMap(x', y') = r * 100 \quad (5)$$

Where $Step_x$ is the length of each grid in Θ -axis direction, whose default value is 0.0314. $Step_y$ is the length of each grid in z -axis direction, whose default value is $(\max(z) - \min(z)) / 100$. The value of $ShapeMap(x', y')$ is 100 times of the value of r then rounded to nearest integer. The reason to magnify its value by 100 times is to reserve the precision to percentile when converted to a gray level value which must be an integer.

2.1.1. Shape map PNG file specification

Figure 6 shows an example of shape map PNG file. It contains two main parts: the caption and shape map value (see figure 7). The caption part contains the file name and the value of $\min(z)$ and $\max(z)$ described in section 2.1. The $ShapeMap(x', y')$ value is stored as the 16 bit gray level value for the PNG file.

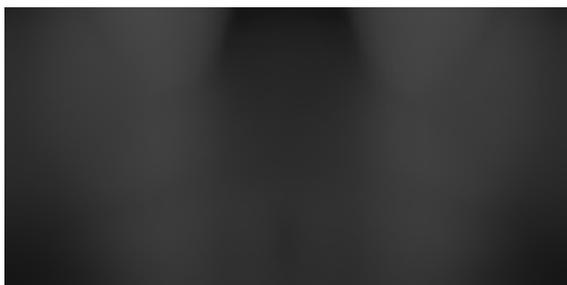


Fig. 6. PNG file: G1_684.8125_946.3625

101	ShapeMap(1,101)	ShapeMap(201,101)
⋮	⋮	⋮
2	⋮	⋮
1	ShapeMap(1,1) value range: [0.65535]	ShapeMap(201,1)
y' \ x'	1	2	3	...	201

Caption: file name_min(z)_max(z)

Fig. 7. Shape map PNG layout

2.2. From shape map data to point-cloud data

To retrieve the body scanning point-cloud data from a shape map PNG file, the caption and shape map value must be read to obtain: $ShapeMap(x', y')$, file name, $\min(z)$, $\max(z)$. Refer to section 2.1, the point-cloud data $Point_{grid}(r, \Theta, z)$ can be obtained as following:

$$r = ShapeMap(x', y') \quad (6)$$

$$\Theta = -\pi + (x' - 1) * Step_x \quad (7)$$

$$z = \min(z) + (y' - 1) * Step_y \quad (8)$$

And the $\text{Point}_{\text{grid}}(x'',y'',z'')$ can be obtained by converting the Polar coordinate's values of $\text{Point}_{\text{grid}}(r,\Theta,z)$ to Cartesian coordinate's values.

3. Results and discussion

We tested the shape map method by converting point-cloud data of waist to hip body section of 75 individuals using the $[\text{TC}]^2$ 3D scanner with the Matlab modules. It took 95.4 seconds to convert the whole set of point-cloud data to shape map data, average 1.3 seconds per sample. While it took 15.2 seconds to retrieve the point-cloud data from the shape map data, average 0.2 seconds per sample on a PC with Inter(R) Core(TM) Duo CPU E8500@ 3.16GHz and 4.00 GB memory installed.

The results show that the shape map PNG files set occupied only 12.0% of the storage room for Point-cloud ASCII text (TXT) files. And it contained more sampling points, which was 3.66 times of the original files averagely. By decreasing the value of Step_x and Step_y , it can increase resolution further, but the size of the shape map PNG file will increase simultaneously.

Table 1. Comparison between point-cloud format and shape map format.

	Point-cloud format	Shape map format
Storage room	16.30 MB (average 0.217 MB per file)	1.95 MB (average 0.026 MB per PNG file)
Sampled point	average 5550 unevenly distributed points	20,301 evenly distributed points

Another advantage for the shape map is that it presents the human scanning file in graphic, which is more intuitive than number. For example, the figure 8 shows two shape map files for sample A and B. It is visualized that the size of A is bigger than B as the overall brightness of A is higher than B, and A has a more obvious belly as the lower middle section of A is brighter compared to the surrounding area.

Refer to sections 2.1 and 2.2, the Gridfit method ensures that the original sample points $\text{Point}_{\text{aligned}}(x,y,z)$ are on the estimated surface, but the original points cannot be accessed directly. It needs to be interpolated using the 4 surrounding $\text{Point}_{\text{grid}}(x'',y'',z'')$.

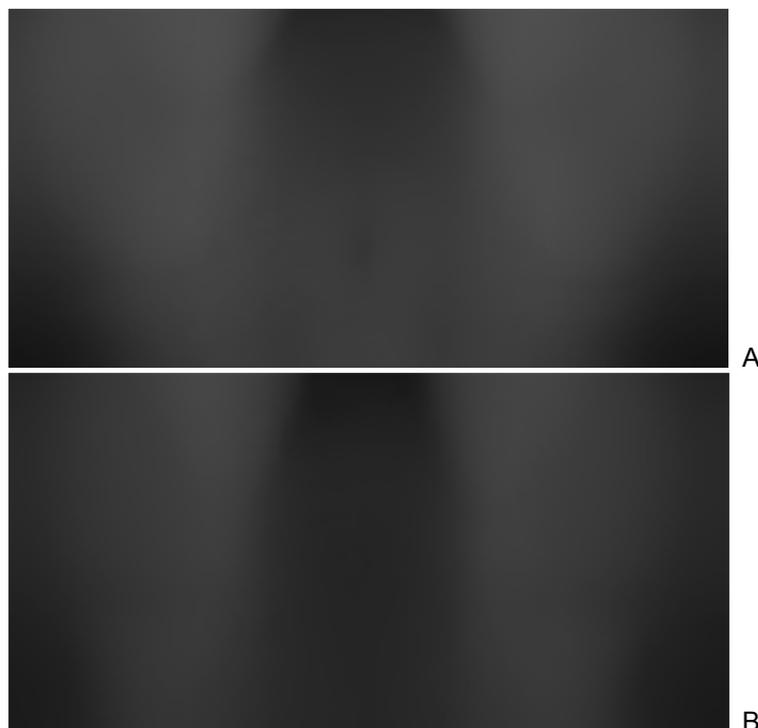


Fig. 8. Two shape map files: A (upper) and B (lower)

4. Future work

The shape map converting module has been developed in Matlab, however it is designed to convert one part of the body each time (see figure 1 right). Our future work involves the design of a higher level module to work with the full body scanning files by joining all the parts back.

Acknowledgement

The authors would like to acknowledge the support from General Research Fund (GRF) projects HK, B- Q26H and The Hong Kong Polytechnic University Funding U894.

References

1. Z. Ping, (2008): "Hip Circumference Is the Critical Horizontal Part That Affects Female Clothes for Heavy-bust and Slim-waist Figures [J]," Journal of Liaodong University (Natural Science), vol. 2.
2. M. Holewun and W. Lotens., (1992): "The influence of backpack design on physical performance," Ergonomics, vol. 35, pp. 149-157.
3. T. Ryu, et al., (2004): "Development and application of a generation method of human models for ergonomic product design in virtual environment," pp. 951-955.
4. S. M. Carcone and P. J. Keir., (2007): "Effects of backrest design on biomechanics and comfort during seated work," Applied ergonomics, vol. 38, pp. 755-764.
5. F. Segovia, et al., (2011): "A comparative study of feature extraction methods for the diagnosis of Alzheimer's disease using the ADNI database," Neurocomputing.
6. T. Sha, et al., (2011): "Feature level analysis for 3D facial expression recognition," Neurocomputing.
7. E. Paquet, (2004): "Exploring anthropometric data through cluster analysis," .
8. M. Petrov, et al., (1998): "Optical 3D digitizers: Bringing life to the virtual world," Computer Graphics and Applications, IEEE, vol. 18, pp. 28-37.
9. T. Weyrich, et al., (2004): "Post-processing of scanned 3D surface data,".
10. J. U. G. D'Errico, 2006. (Accessed 2009): <http://www.mathworks.com/matlabcentral/fileexchange/8998>.