Meshes vs. Depth Maps in Face Recognition Systems

Sebastian Pabiasz and Janusz T. Starczewski

Department of Computer Engineering,
Czestochowa University of Technology, Czestochowa, Poland
{sebastian.pabiasz,janusz.starczewski}@kik.pcz.pl

Abstract. The goal of this paper is to present data structures in 3D face recognition systems emphasizing the role of meshes and depth maps. 3D face recognition systems are still in development since they use different data structures. There is no standarized form of 3D face data. Dedicated hardware (3D scanners) usually provide depth maps of objects, which is not sufficiently flexible data sturcture. Meshes are huge structures and operating on them is difficult and requieres a lot of resources. In this paper, we present advantages and disadvantages of both types of data structures in 3d face recognition systems.

Keywords: biometric, 3D face, mesh, depth map.

1 Introduction

Over the past 20 years many different face recognition techniques were proposed. Originally they were based on flat images. Turk and Pentland [1] used eigenfaces for face detection and identification. Zhao et al. [2] presented the use of the LDA algorithm in face recognition. A classification method using the SVM technique was presented by Heisele et al.[3]; they developed a whole set of tools for processing and comparing facial photographs. However, these solutions do not work well in applications where reliability is a priority. The solution may be systems based on 3D data. In 2D facial biometrics research, the material is provided by a photography, which is a basic data structure. In 3D facial biometrics, we have few data structures, such as normals maps, depth maps, meshes and corresponding to them photographs. Such a number of different types of data makes the system difficult to operates on them. Already, there exist a number of public methods for 3D reconstruction (e.g. [4], [5], [6]). Depending on the method we have a limited number of data structures. A typical 3D scanner gives us a depth map so thus we can generate a mesh based on it. Other methods, that use a flat as a data source give us differential data structures.

2 Data in 3D Biometric

In 3D face biometrics, we distinguish multiple types of data. The kind of them depends on the method of acquisition. Typical dedicated hardware, like 3D

L. Rutkowski et al. (Eds.): ICAISC 2012, Part I, LNCS 7267, pp. 567–573, 2012.

scanner, give us depth map of surface, which is our basic structure that we can generate a mesh based on it. An alternative method is to use photometry stereo to reconstruct a face from flat images. In this case, our basic structure is a map of normals, from which we can generate a depth map and then a mesh.

2.1 From Normals to Depth Map

Methods based on bidirectional transmittance distribution function (BRTF, e.g. [7], [8]), as a first result, give a map of normals of an object. To build a depth map, we apply the least-squares technique. We use the fact that if the normal is perpendicular to the surface, then it will be perpendicular to any vector on it. We can construct vectors on the surface using neighbor pixels, more precisely, we employ a pixel from the right and below. The normal at a surface must be orthogonal to the vector Vec1, i.e.,

$$\begin{aligned}
&Vec1 = (x+1, y, Z_{x+1,y} - (x, y, Z_{x,y}))\\
&Vec1 = (1, 0, z_{x+1,y} - Z_{x,y})\\
&Normal.Vec1 = 0\\
&(N_x, N_y, N_z).(1, 0, Z_{x+1,y} - Z_{x,y}) = 0
\end{aligned}$$

$$N_x + N_z(Z_{x+1,y} - Z_{x,y}) = 0$$

and to the vector Vec2 leads to

$$\begin{aligned}
&Vec2 = (x, y+1, Z_{x,y+1} - (x, y, Z_{x,y}))\\
&Vec2 = (0, 1, Z_{x,y+1} - Z_{x,y})\\
&Normal.Vec2 = 0\\
&(N_x, N_y, N_z).(0, 1, Z_{x,y+1} - Z_{x,y}) = 0
\end{aligned}$$

$$N_y + N_z(Z_{x,y+1} - Z_{x,y}) = 0$$

If the pixel does not belong to an object, we describe this as

$$-N_x + N_z(Z_{x-1,y} - Z_{x,y}) = 0 \tag{1}$$

$$-N_y + N_z(Z_{x,y-1} - Z_{x,y}) = 0 \tag{2}$$

For each pixel of an object, we construct two entries so our main matrix will be of the form $M(2*number\_of\_pixels, number\_of\_pixels)$. We can express our matrix equation as $MZ = Vec$, but the least squares method solves the equation $M^TMZ = M^TVec$, so thus the main matrix M^TM will be extremely big.

2.2 Depth Map

A depth map is a matrix, which elements represent the height of a particular pixel. Fig.1 presents an objects which depth map is in Table 1. In this case, resolution is 8x8. 3D scanners usually deliver data in the form of a depth map, additionally they can deliver information which pixel is valid (belongs to the object).

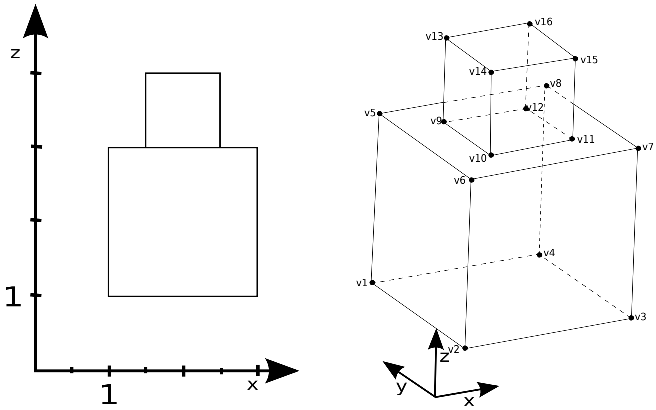

Fig. 1. Exaple of a 3D picture

Table 1. Depth map of figure from Fig.1

0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	3	3	3	3	0	0
0	0	3	4	4	3	0	0
0	0	3	4	4	3	0	0
0	0	3	3	3	3	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

2.3 Meshes

Meshes are more complex data structures than depth maps. The mesh is usually a collection of vertices, edges and faces. The face usually consist of triangles or quadrilaterals.

Table 2. Coordinates of verticles from Fig.1

Number	x	y	z	Number	x	y	z
1	1	-1	1	9	0.5	-0.5	3
2	1	1	1	10	0.5	0.5	3
3	3	1	1	11	2.5	-0.5	3
4	3	-1	1	12	2.5	0.5	3
5	1	-1	3	13	0.5	-0.5	4
6	1	1	3	14	0.5	0.5	4
7	3	1	3	15	2.5	-0.5	4
8	3	-1	3	16	2.5	0.5	4

Table 3. Table of faces from Fig.1

Number	v1	v2	v3	v4	Number	v1	v2	v3	v4
1	1	2	3	4	7	9	10	11	12
2	1	2	6	7	8	9	10	14	13
3	2	3	7	6	9	10	11	15	14
4	3	4	8	7	10	11	12	15	16
5	4	1	5	8	11	12	9	13	16
6	5	6	7	8	12	13	14	15	16

Table 2 and 3 present basic mesh structures for the object from Fig. 1. In this case, squares were used as the basic faces shape.

2.4 Building a Mesh from a Depth Map

The depth map is the main data source for building the mesh. First step in this process is to select such points that are basic to build faces. We also have to make a decision about a shape of the faces (usually triangles are used). Use all points from a depth map, the resultant mesh can be extremely big and can require large resources and time. To reduce the number of the basic points, we can apply a simply algorithm (Alg. 1).

Algorithm 1. Depth map reduction algorithm

```
for x = 1 → MAXX do
  for y = 1 → MAXY do
    if i = factor then
      reduced[x, y] ⇐ source[x, y]
      i ⇐ 0
    else
      i ⇐ i + 1
    end if
  end for
end for
```

The results of Algorithm 1 are presented in Fig. 2. As we can observe, this solution reduce the mesh uniformly. Some important points may be lost. A significant improvement of quality can be obtained by key selection points and reduction. Key points can be obtained by filtration. Mesh construction is, in this case, difficult because a depth map is not uniformly reduced. Algorithm 2 presents our solution to build mesh from a uniform depth map, and Fig. 3 presents results of reduction with factor 20 and key point selection.

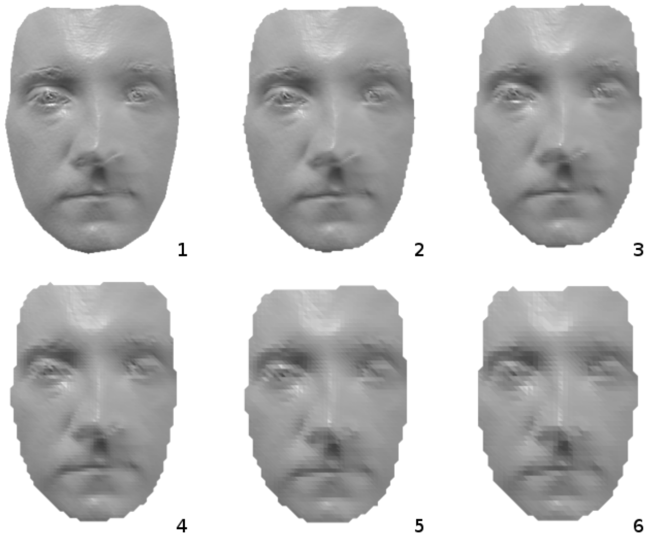

Fig. 2. Reduction factor (1-6)

Algorithm 2. Building mesh from uniforlmy depth map

for $x = 1 \rightarrow MAXX$ **do**
  for $y = 1 \rightarrow MAXY$ **do**
    findSection {2 points y1 and y2}
    $findPoints(x - 1)$
    $findPoints(x + 1)$ {In lines x-1 and x+1 from y1 to y2}
    for $point = 1tofoundedPoints$ **do**
      $buildTriangle(y1, y2, point)$
    end for
  end for
end for

3 Comparison

Comparison is the most important process in verification and identification of faces. In the past years, many solutions have been proposed (e.g. [9], [10], [11]). They operate on different principles and are effective to varying degrees.

3.1 Depth Map vs. Depth Map

The comparison of two depth maps directly is difficult because values in one cell on the first depth maps must correspond exactly to the same cell on the second depth map. The process of acquisition must be very accurate, what is often feasible. The second solution is to calibrate depth maps based on landmarks.

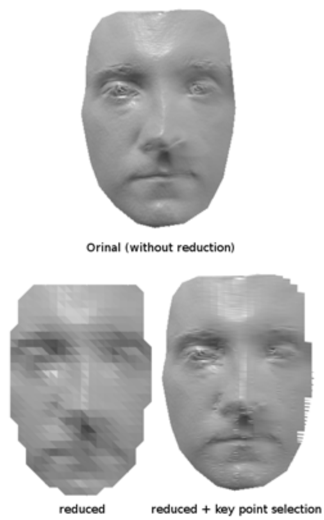

Fig. 3. Reduction with the key points selection

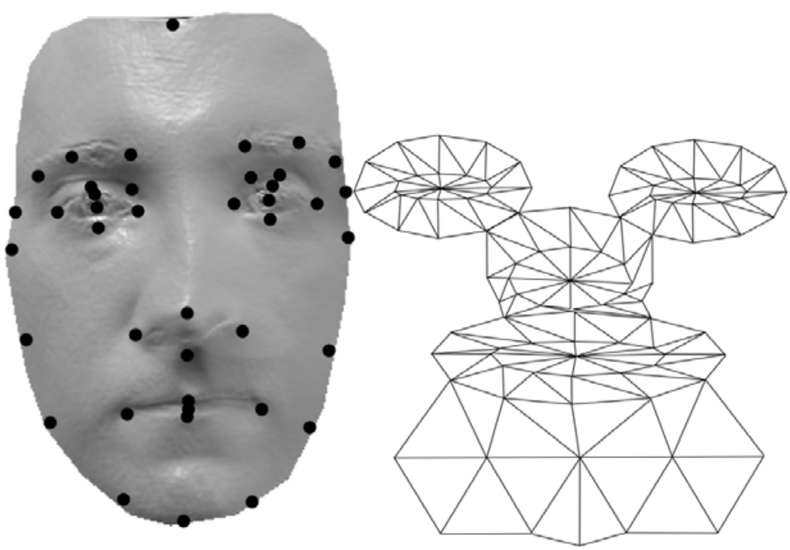

Fig. 4. Mesh based on landmarks

It is much more better solution; however, we must take into attention errors of acquisition apparatus. Summarizing, obtaining two identical maps of the same object is very difficult and laborious.

3.2 Mesh vs. Mesh

A medium-sized facial mesh consists of about 50000 points. The comparison of two meshes in this form is very resource-intensive and time consuming. As in the case of depth maps, calibration is required; however, to perform the comparison is much easier since we compare two faces which need not coincide perfectly. Fig. 3 presents a mesh build on landmarks, which results in that the number of points is always constant.

4 Final Remarks

Depth maps and meshes are inextricably connected data structures, they complement each other. In 3D biometric systems, we can use intelligently reduced meshes, which are more flexible and they are not restrictive as depth maps in comparison. In this area, is still much work to do.

References

1. Turk, M.A., Pentland, A.P.: Face recognition using eigenface. In: Internat. Conf. on Pattern Recognition, pp. 586–592 (1991)
2. Zhao, W., Chellappa, R., Krishnaswamy, A.: Discriminant analysis of principal component for face recognition. In: 3rd IEEE Internat. Conf. on Automatic Face and Gesture Recognition, pp. 336–341 (1998)
3. Heisele, B., Ho, P., Poggio, T.: Face recognition with support vector machines: Global versus component-based approach. In: 8th IEEE Internat. Conf. on Computer Vision, pp. 668–694 (2001)
4. Levine, M.D., Yu, Y.: State-of-the-art of 3D facial reconstruction methods for face recognition based on a single 3D training image per person. Pattern Recognition Letters 30, 908–913 (2009)
5. Mahoor, M.H., Abdel-Mottaled, M.: Face recognition based on 3D ridge images obtained from range data. Pattern Recognition 42, 445–451 (2009)
6. Jiang, D., Hu, Y., Yan, S., Zhang, L., Zhang, H., Gao, W.: Efficient 3D reconstruction for face recognition. Pattern Recognition 38, 787–798 (2005)
7. Song, P., Wu, X., Wang, M.Y.: Volumetric stereo and silhouette fusion for image-based modeling. Vis. Comput. 26, 1435–1450 (2010)
8. Pabiasz, S., Starczewski, J.T.: Face reconstruction for 3d systems. Selected Topics in Computer Science Applications, 54–64 (2011)
9. Huang, J., Blanz, V., Heisele, B.: Face recognition with support vector machines and 3D head models. In: Internat. Workshop on Pattern Recognition with SVM, pp. 334–341 (2002)
10. Blanz, V., Vetter, T.: Face recognition based on fitting a 3D morphable model. IEEE Transaction on Pattern Analysis and Machine Intelligence, 1063–1074 (2003)
11. Lu, X., Zhang, H.: Adapting geometric attributes for expression-invariant 3D face recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence 30, 1346–1357 (2008)