DOI: 10.24996/ ijs.2017.58.3C.17

ISSN: 0067-2904

Surface Shape Descriptors on 3D Faces

Shaimaa H. Shaker[1*], Najlaa Abd Hamza[2]
[1] Department of Computer Sciences, University of Technology, Baghdad, Iraq.
[2] College of Nursing, University of Baghdad, Baghdad, Iraq.

Abstract
The general objective of surface shape descriptors techniques is to categorize several surface shapes from collection data. Gaussian (K) and Mean (H) curvatures are the most broadly utilized indicators for surface shape characterization in collection image analysis. This paper explains the details of some descriptions (K and H), The discriminating power of 3D descriptors taken away from 3D surfaces (faces) is analyzed and present the experiment results of applying these descriptions on 3D face (with polygon mesh and point cloud representations). The results shows that Gaussian and Mean curvatures are important to discover unique points on the 3d surface (face) and the experiment result shows that these curvatures are very useful for some special features extraction for eyes and nose detection.

Keywords: 3D Face, Surface Shape Descriptor, Gaussian Curvature, Mean Curvature.

واصفات شكل السطح للوجوه ثلاثية الابعاد

شيماء حميد شاكر[1*]، نجلاء عبد حمزة[2]
[1] قسم علوم الحاسبات،الجامعة التكنلوجية، بغداد، العراق.
[2] كلية التمريض ، جامعة بغداد، بغداد، العراق.

الخلاصة
ان الهدف العام من تقنيات وصف شكل السطح هو لتصنيف العديد من الأشكال السطحية من مجموعةا لبيانات. انحناءات غاوس (K) و مين (H) هي المؤشرات الأكثر استخداما على نطاق واسع لتوصيف (تمييز) شكل السطح في تحليل مجموعة من الصور.في هذا البحث شرح مفصل عن هذه المؤشرات(Hو K)، وقد تم تحليل القوة التمييزية لتلك الواصفات ثلاثية الأبعاد المأخوذة من الأسطح ثلاثية الأبعاد (وجوه) وتقديم نتائج التجربة لتطبيق هذه الأوصاف على الوجه ثلاثي الأبعاد (مع شبكة مضلعة والتمثيلات السحابية النقطية). النتائج وضحت ان منحنيات غاوس ومين هي جدا مهمة في كشف النقاط الفريدة (المميزة) في السطح ثلاثي الابعاد (الوجوه) ، وايضا بينت النتائج انها جدا مهمة في استخلاص بعض الخصائص من الوجوه مثل تحديد منطقة العين والوجه.

1. Introduction
Local shape descriptors can be used for a variety of activities, from registration to correlation with shape analysis and recovery, it's used in 3D shape corresponding to discovery distinctive points on the

*Email: sh_n_s2004@yahoo.com

surface. They can also be utilized to speed up seeking by decreasing the model to a little amount of features which are simply matched [1].

Ideal local descriptor is invariant to "insignificant" geometrical variations, classically rotation and translation, occasionally scaling, and bending. It should also be robust to noise and geometrical noise (vertices moving), Mesh topology noise (the mesh connectivity alterations), also, general topology noise (the formation of handles and passages). To treat invariance, most descriptors quantity geometrical feature that are invariant to interpretation and alternation, such as curvature, length, volume, and angle [1].

The goal of this paper is explain the details of some surface shape descriptors (K and H), and present the experiment results of applying these descriptions on 3D face (with polygon mesh and point cloud representations).

In the rest of this paper, show a relevant literature on Surface Shape Descriptors (K and H) on 3D surface (Face) in section 2, display primary 3D surface representations in section 3, Section 4 shows a method of mesh storage, Section 5 shows theoretic concepts of surface curvature, Section 6 displays experiment results and Analysis, Section 7 shows Conclusion, and, finally, a list of the references in section 8.

2. Literature Survey

In this section, a relevant literature on Surface Shape Descriptors (K and H) on 3D surface (Face) has been reviewed.

The work presented in [2] contain clarify how compute the curvature, and comparison with other attributes (Dip angle attribute and Azimuth attribute) and conclude that is the curvature (K and H) is better at identifying surface and it is surface direction independent, which implies that the curvature value remains not changed, regardless of the possibility that the surface is rotated or tilted.

In [3] Displayed an method for 3D face recognition from range information in view of the mark of curvature(H and K), utilize the principal curvature, k_{max}, to characterize the face shape as a 3D binary image termed ridge image. The ridge image demonstrates the areas of the edge lines around the essential facial locales on the face (i.e. the eyes, the nose, and the mouth).

The work in [4], includes a face recognition framework utilizing a HK division (situated in the indications of the H and K curvature) for isolating region of explained curvature has been performed.
The work submitted in [5] proposed a technique, the Curvature Map(H and K Curvature), that utilizations surface curvature properties in an area around a point make a extraordinary mark for that point. These marks can then be matched to define the correspondence of one point to another.

3. Primary Representations of 3D Surface

The massive popular of algorithms in graphics and computer vision work on 3D representations constructed at surfaces [6], there are variety kinds of 3D surface representations. In this section, explains Point Cloud representation and 3D Mesh representation.

1. Representation of Point Cloud

A Representation of Point Cloud as shown in Figure-1 is simply a combination of the *z*, *x* and *y* coordinates, every one identify a 3D point at the point cloud (3D surface). Let $c = (x, y, z) \in R^3$ mean a 3D point in 3D surface *C*.
Instead, a 3D point can be denoted as vector of 3D point, and 3× N matrix represented as a point cloud **C**, wherever *N* is the points number [7].

$$c = [x \quad y \quad z]^T \qquad (1)$$

$$P = [p1 \dots pN] \qquad (2)$$

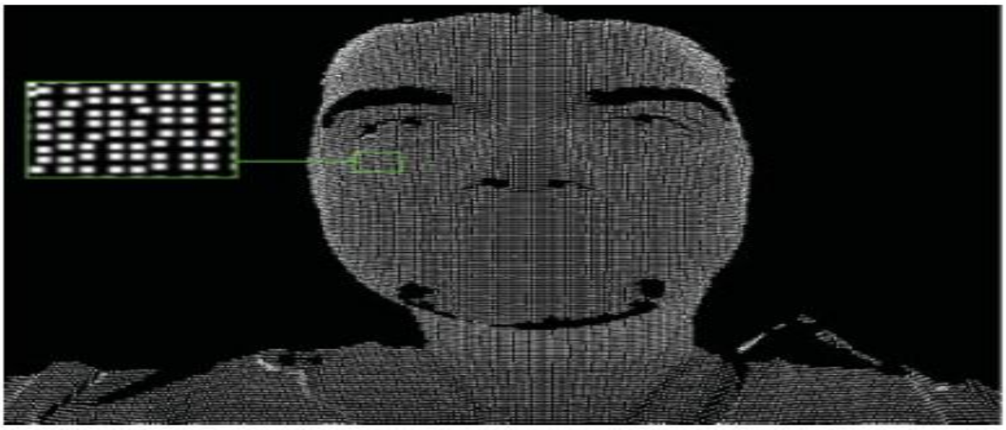

Figure 1- Point cloud representation [7].

2. 3D Mesh Representation

A 3D mesh N denotes 3D surface utilize groups of mesh components, vertices T, edges D, and polygons (facets) G, $N = (T, D, G)$. The vertices of mesh represent 3D location, $T \subset R^3$. Every edge, $d_i \in D$, is distinct through two separate vertices, $D \subset \{d_i = \{t_j, t_k\} \mid t_j , t_k \in T, j \neq k\}$. Whereas every facet, $g_i \in G$, is unmistakable by at least three edges with the end goal that each couples of edges share a vertex. (The vertex is event to together edges.)

In the situation of a triangular mesh as shown in Figure -2 (which is the greatest broadly used kind of mesh consequent its relation straightforwardness), the facet d_i is precisely characterized by three edges $G \subset \{g_i = \{d_m , d_b , d_p \} \mid d_m , d_b , d_p \in D, m \neq b, m \neq p, m \neq p, \forall d_1, d_2 \in g_i \ \exists t \ (t \in d_1 \wedge t \in d_2)\}$. Instead, every facet is distinct through three different vertices, $G \subset \{ g_i = \{t_m, t_b , t_p \} \mid t_m , t_b , t_p \in T, m \neq b, m \neq p, b \neq p\}$.

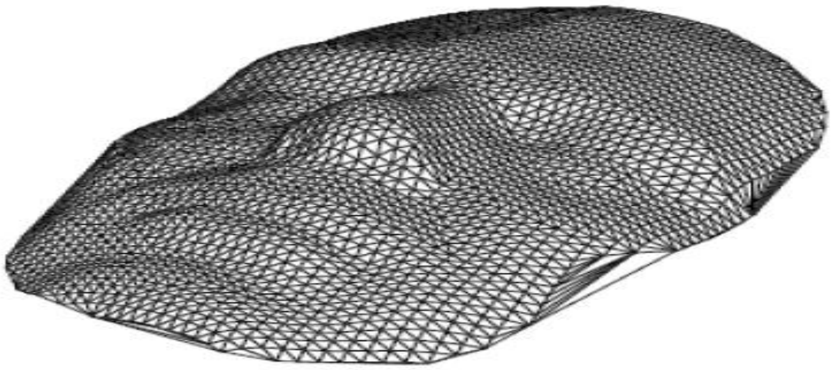

Figure 2- Mesh of facial surface [6].

4. Storage of Mesh

There are an extensive variety of restrictive and open file styles for the mesh information storage. These may be classified into ASCII text and binary formats. The previous are editable and intelligible while the last are more space effective. In generic, these file styles all contain a listing of vertices and a listing of polygons that indicator to the vertices list, The files can be likewise store face or vertex properties, for example, surface normal or coordinates of texture.

The greatest frequently used formats based on text are OBJ (formerly advanced by Virtual Reality Modeling Language (VRML) and Wavefront Technologies), which was planned especially with the World Wide Web in opinion. Lastly, the Polygon (ply) File Format (established at the Stanford Graphics Lab) shore together binary and ASCII storage [1].

In next section, briefly define the PLY format.

1. Polygon (ply) File Format

The polygon (ply) file format (stanford triangle style), was proposed with the end goal of being both an adaptable and convenient 3D file styles. It has together a binary and an ASCII form. The form of binary contains data to make it tool separate, indicating the numeral of bytes each kind, the types utilized to every value, and if it's huge or small endian. Furthermore, the style lets to user determine kinds permitting it to become extensible for necessities of aftertime 3D information. Due to its straightforwardness and adaptability the polygon (ply) file format is exceptionally prominent in the research and scholarly world. A file of ply starts with header of text as:

```
ply format ascii 1.0
comment author: John Smith
element vertex 3
property float x
property float y
property float z
element face 1
property list uchar int vertex_index
end_header
```

When the principal bar distinguishing the file is a ply format the header holds various types , keys, and accounts determining the file layout. The "format" word classifies that file as binary or text founded. Taking after that are various "element" words, in this illustration characterizing the vertices layout initially, thereafter faces. Particularly the bar "element vertex 3" expresses that the file holds three vertices. Taking after a "element" words are various "property" words. These attributes characterize what precisely a vertex is. For this situation, it shows a vertex hold three sections, a z, x, and y part, and every of part is spoken to by a value of floating point. "end header" word expresses that the finish of the header. The genuine data takes after the finish of header, For instance a body running permitting with the above header may resemble:

```
0.0 0.0 0.0
0.0 1.0 0.0
1.0 0.0 0.0
3 0 1 2
```

The polygon (ply) format shore: geometry with the formula of faces/ edges/ vertices, textures , materials , and vertex colors [8].

5. Theoretic Concepts of Surface Curvature

The curvature idea includes an estimation of drooping quantitative of the curve at specific point. What's more, it gauges characteristic feature of the surface of facial, taking in thought its imperviousness to rigid converting. Hypothetically, the data of curvature is computed as the reverse radius *R* of the osculating circle that topically suit the curve *C*. consequential, a circle has a fixed evenly curvature account more than zero, whereas a straight line T include a zero curvature account as its osculating circle hold its center to infinity as demonstrated in Figure-3 [6].

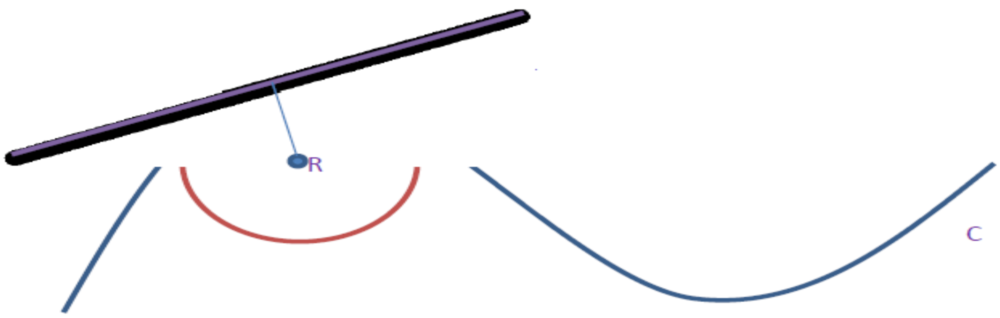

Figure 3- Hypothetical concept of surface curvature in 2D [6].

1. Curvatures of 3D Surfaces

Assume *o* a point on a smooth surface *U*, assume *No* the unit normal to *U* at *o*, and assume *X* (*e*, *f*) is a local parameter of *U* in a neighbourhood of *o*. Thereafter with *Xe*(*o*), X*f*(*o*), *No* as a local coordinate structure, principal directions and principal curvatures will calculate as follows: assume *y*1 and *y*2 (*y*1 ≥ *y*2) be the eigenvalues, and *o*1, *o*2 the related unit eigenvectors of the Weingarten curvature matrix

$$W = \begin{pmatrix} \frac{uW - vV}{UW - V^2} & \frac{vU - uV}{EG - F^2} \\ \frac{vW - wV}{UW - V^2} & \frac{wU - vV}{UW - V^2} \end{pmatrix} \dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots (1)$$

Where

$$u = N_o . X_{ee}(o) \qquad U = X_e(o).X_e(o)$$
$$v = N_o . X_{ef}(o) \qquad V = X_e(o).X_f(o)$$
$$w = N_o . X_{ff}(o) \qquad W = X_e(o).X_f(o)$$

Take note of that in the exceptional case that $X_e$and X_f are orthogonal unit vectors, this turns into the symmetric matrix

$$W = \begin{pmatrix} u & v \\ v & w \end{pmatrix} \dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots (2)$$

In the event that *e* is a unit vector in the tangent plane to *U* at *o*, then

$k_e = e^T We \ \dots\dots (3)$

is the normal curvature of the surface toward of e.

It takes after that y2 and y1 are the minimum and maximum normal curvatures of the surface (u) at o, and $o1 = \binom{o11}{o12}$ and $o2 = \binom{o21}{o22}$ are the principal curvature vectors communicated in local coordinates. That is,

$$f_1 = o_{11}X_e + o_{12}X_f$$
$$f_2 = o_{21}X_f + o_{22}X_f \quad \dots\dots\dots(4)$$

are the principal direction vectors in *R*3[9].

2. Surface Shape Descriptors founded from Curvature Data:

The main surface shape descriptors applied in this work are exhibited in the next subsections, which are set up from principal curvaturep_1, p_2[6].

Gaussian Curvature (*K*):

K=p_1*p_2 ……………………………………………………………………………………(5)

The indication of K by a point p at a surface S has a vital geometric significance, which is in detail beneath.

1. K > 0 The principal curvatures p_1 and p_2 have the same sign. The normal curvature k_u in any tangent direction t is equal to $p_1 cos^2\theta + p_2 sin^2\theta$, where θ is the angle between the principal vector corresponding to $p_1$and t. Thus p has the equal indication as that of p_1 and p_2. The surface is curvature far from its tangent plane in all tangent bearings at p. The quadratic estimate of the surface close p is the parabolic. We summon p an **elliptic point** of the surface. Figure -4 shows an elliptic parabolic z = x^2+ $2y^2$ with principal curvatures 2 and 4 at the origin.

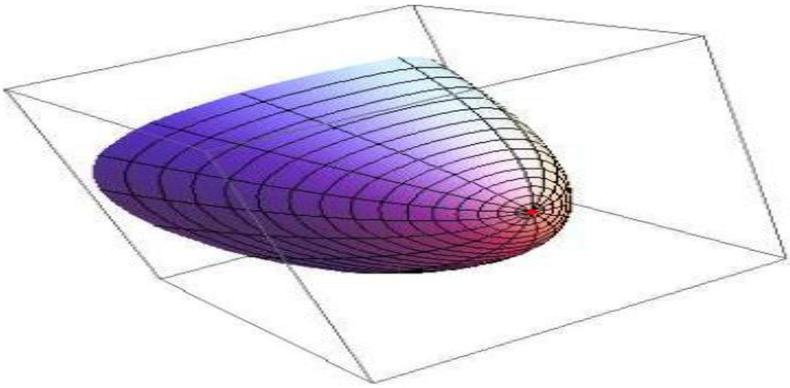

Figure 4- Elliptic point [10].

2. K < 0 The principal curvatures p_1 and p_2 have inverse signs at p. The quadratic estimate of the surface nearby p is a hyperboloid. The point is assumed to be a hyperbolic point of the surface. Figure -5 demonstrates a hyperbolic parabolic z = x^2+ $2y^2$ with principal curvatures 2 and −4 at the origin.

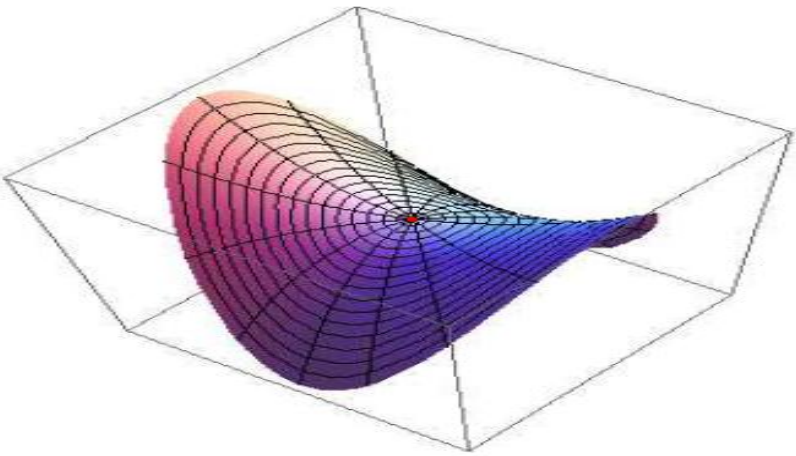

Figure 5-Hyperbolic point.

3. K = 0 There are two cases:

a. Just a single principal curvature, say, p_1, is zero. For this situation, the quadratic estimate is the cylinder $p = \frac{1}{2}p_2y^2$. The point p is named a parabolic point of the surface.

b. Both principal curvatures are zero. The quadratic estimation is the plane z = 0. The point p is a planar point of the surface. One cannot decide the nature of the surface nearby p without analyzing the third or higher order derivatives. For instance, a point in the plane and the origin of a monkey saddle z = $x^2 - 3xy^2$ Figure-6 are both planar points, yet they have very unique shapes.

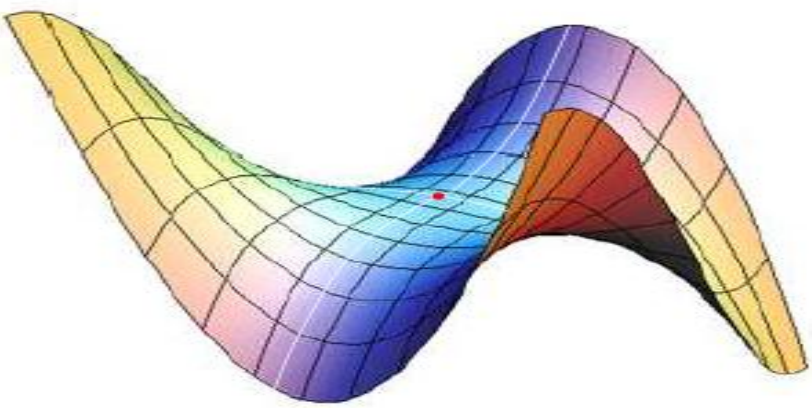

Figure 6- Planar point [10].

Mean curvature (H):

$$H=\frac{p_1+p_2}{2} \quad \ldots\ldots(6)$$

Many shapes cannot be distinguished by *K* only, yet require the expansion of the mean curvature information. The discovery of local shape from this mixture of curvatures is compressed in Table -1 and Figure-7.

Table 1- Curvature shape classification [11]

	K<0	K=0	K>0
H<0	Saddle Valley	Concave Cylinder	Concave Ellipsoid
H=0	Minimal	Plane	Impossible
H>0	Saddle Ridge	Convex Cylinder	Convex Ellipsoid

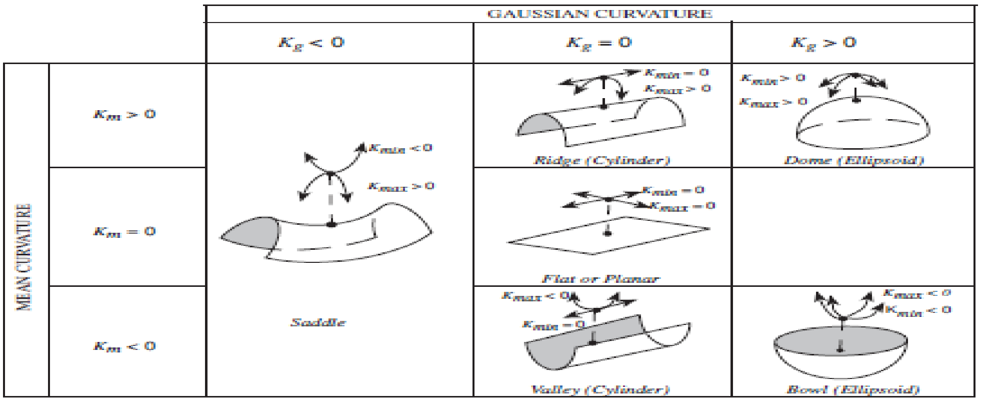

Figure 7- Curvature shape organization [12].

6. Experiment Results and Analysis:

In this section experimental results are presented after applying the descriptions (*H* and *K*) on 3D face (represent by polygonal mesh and point cloud formats) as shown in Figures- (8, 9).

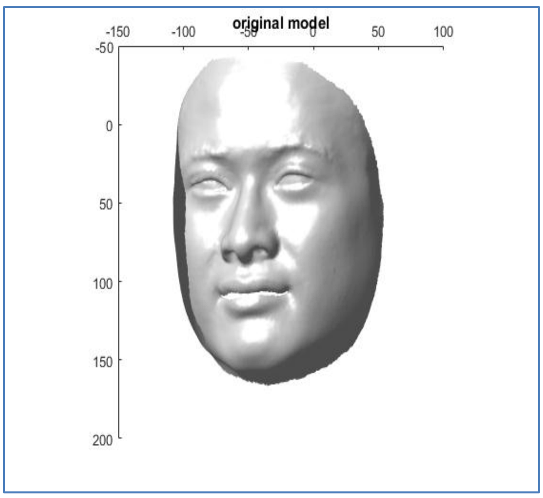

Figure 8- 3D model.ply

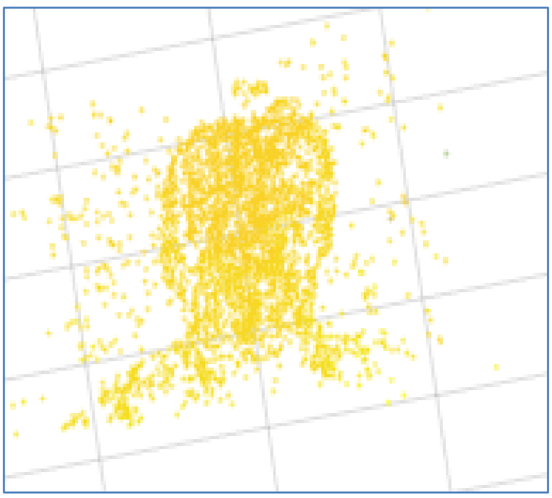

Figure 9- 3D point cloud

In the following the main steps needed for calculate *K* and *H* curvatures are explained.

a- Read 3D Cover Model

There are many different 3D model file formats, in this paper, one type is used, '.ply' which is the most common 3D file content format. Algorithm (1) contains the main steps needed to read 3D model file.

b- apply *K* and *H* on 3D Model

The main steps of the method to calculate *K* and *H* curvatures are stated in the algorithm (2).

Algorithm(1): Calculate of Mean and Gaussian curvature
Input: 3D file name **Output: Faces array (F) ,Vertices array(V)**

Begin
Filename= 3D file name
If not empty (Filename)
set (V) **//Read the vertices size from 3D file**
No of vertices= size (V)
For each vertex Vi in V array do { where 1≤ i ≤ No of vertices }
Read (Vi)
Endfor
set (F) **// Read the faces size from 3D file**
No of faces=size (F)
For each face Fi in F array do { where 1≤ i ≤ No of faces }
Read (Fi)
Endfor
Endif
End

Algorithm(2): Calculate of Mean and Gaussian curvature
Input: 3d model of .ply (or 3d point cloud) Output: Mean and Gaussian curvature values
Begin Read 3d model (by using algorithm 1) Calculate Gaussian curvature by using equation (5) Calculate mean curvature by using equation (6) Print value of Mean and Gaussian curvature Plot –mesh (saturation of mesh with better contrast dependent on value of Mean and Gaussian curvature) End

The following figures demonstrate experimental results of preformation algorithm.

Figure -10 shows the results of *K* values on face partitions and how the different reigns of the face are surrounded with certain curves and colors, the same with H values in Figure-11. Marks of the (*K*) and (*H*) curvatures are useful to perform point grouping in collection images as per the neighborhood shape nearby every point. This permits the classification of the diverse patches of surface in a 3D face as indicated by their shape, in light of variance geometry. The 3D regions of surface are categorized as convex elliptical, concave elliptical, concave cylindrical, planar, hyperbolic, or convex cylindrical.

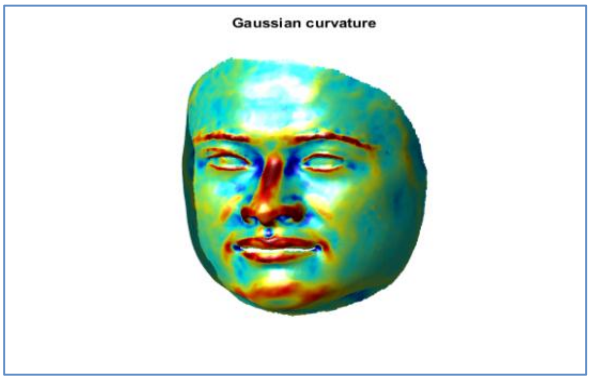

Figure 10- Gaussian Curvature

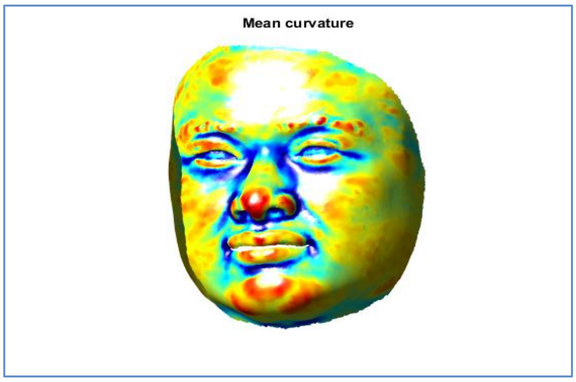

Figure 11-Mean Curvature

For the 3D model.ply, Figure-12 shows *K* values ranging from -0.02 to 0.06, the positive values more than the negative ones and the histogram takes the right more than left, when the points values between (0 to 3500), while *H* values take the left side of the histogram so that negative values more than positives ranging from 0 to 3000, while the points values between (-0.06 to 0.03), in Figure-13. and for 3D point cloud, Figure -14 shows *K* values ranging from -0.2 to 0.2, when the points values between (0 to 1350), while *H* values ranging from 0 to 1950, while the points values between (-0.5 to 0.1), in Figure-15.

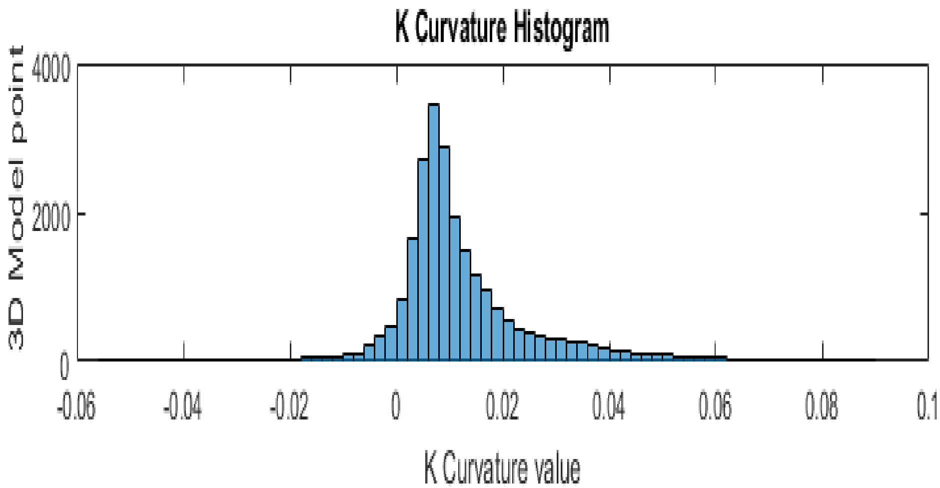

Figure 12- Histogram of Gaussian curvature for 3D model.ply.

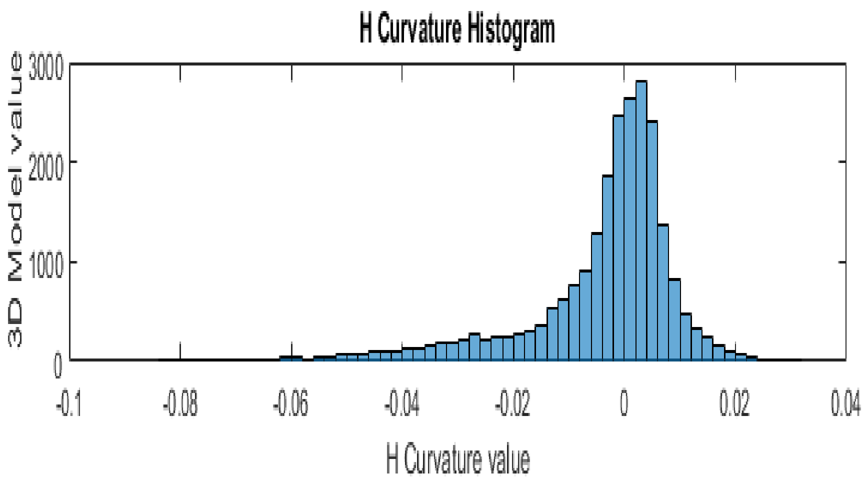

Figure 13- Histogram of Mean curvature for 3D model.ply

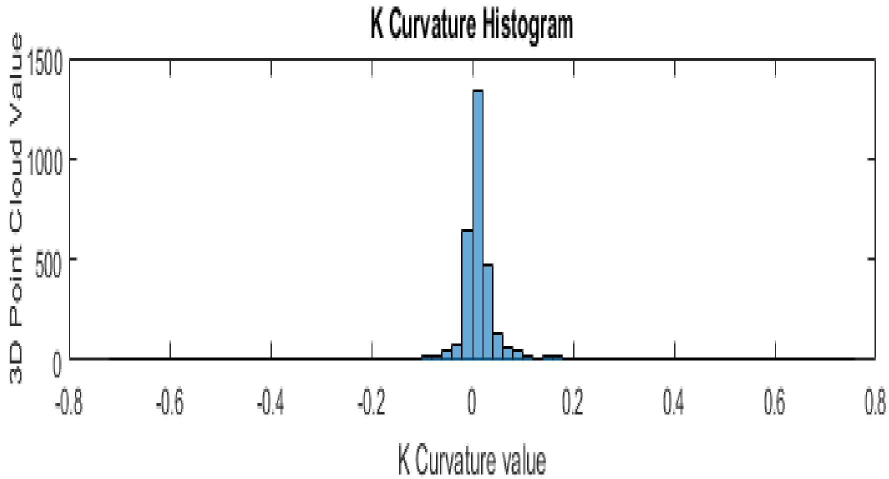

Figure 14-Gaussian curvature histogram for 3D point cloud.

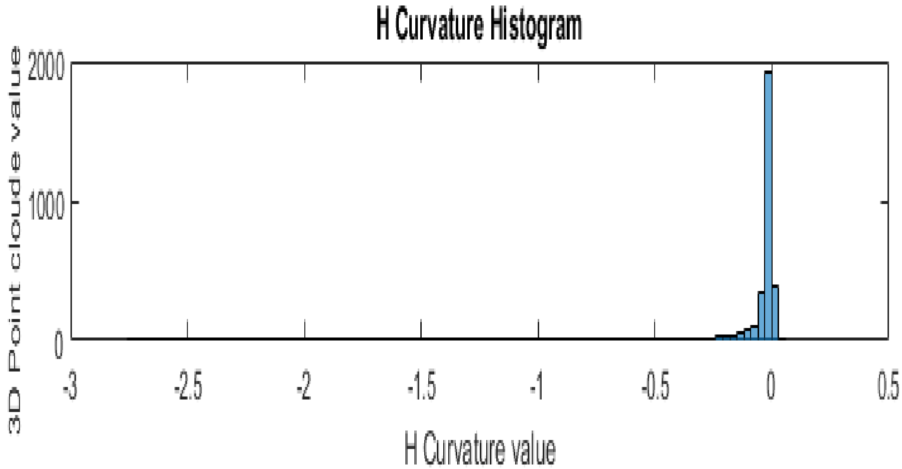

Figure 15- Mean curvature histogram for 3D point cloud.

Conclusion*:*

From the experiment results, the surface shape descriptors techniques can be utilized in 3D shape (face) to discovery unique (distinctive) points on the surface and to categorize several surface shapes from collection data. Mean and Gaussian curvatures are the greatest broadly utilized pointers for surface shape representation in 3D face analysis, and for face segmentation based on these curvatures.

References

1. Heider, P., Pierre, A., Ruosi, L., and Grimm, C. **2011**. Local shape descriptors, a survey and evaluation, 3DOR'11 Proceedings of the 4th E Eurographics conference on 3D Object Retrieval, Llandudno, UK, 49-56, doi:10.2312/3DOR/3DOR11/049-056 .
2. Cantzler, H. and Fisher, R. B. **2001**. Comparison of HK and SC curvature description methods. Proceedings Third International Conference on 3-D Digital Imaging and Modeling, Quebec City.
3. Mahoor, M. H. and Abdel-Mottaleb, M. **2007**. 3D Face Recognition Based On 3d Ridge Lines In Range Data**.** *IEEE Xplore, San Antonio, TX, USA*, 137-140. doi: 10.1109/ICIP.2007.4378910.
4. Moreno, A. B. Á. Sánchez, J. F. and Díaz, F. J. **2004**. Face recognition using 3D surface-extracted descriptors. Machine Vision and Image Processing Conference Sepetember, Universidad Rey Juan Carlos.
5. Gatzke, T. Grimm, C. Garland M. and Zelinka, S. **2005**. Curvature Maps For Local Shape Comparison**.** SMI'05 proceedings of the International Conference on Shape Modeling and Applicatios. Washington, DC, USA: 246-255.
6. Pears, N. Liu, Y. and Bunting, P. **2012**. *3D Imaging, Analysis and Applications*. Springer-Verlag London, E-Book.

7. Daoudi, M. Srivastava , A. and Veltkamp**,** R. **2013**. *3D Face Modeling, Analysis And Recognition***.** Wiley & Sons Ltd, New Delhi, India, E- Book .

8. Mahmood, S. A. **2014.** Pose Invariant Nose Region Detector for 3D Face Recognition**.** Ph.D. Thesis, Department of Computer Sciences, University of Technology.

9. McHenry, K. and Bajcsy, P. **2008**. An Overview of 3D Data Content, File Formats and Viewers, Technical Report. National Center for Supercomputing Applications, University of Illinois at Urbana-Champaign, Urbana.

10. Goldfeather, J. And Interrante, V. **2004**. A Novel Cubic-Order Algorithm for Approximating Principal Direction Vectors, ACM Transactions on Graphics, New York, NY, USA, **23**(1): 45–63. doi: 10.1145/966131.966134.

11. Roberts, A. **2000**. Curvature Attributes and their Application to 3D Interpreted Horizons. Wiley, **19**(2): 85–100. doi**:** 10.1046/j.0263-5046.2001.00142.x

12. Yan-Bin, J. **2001**.*Gaussian Curvature.In: Andrew, P. (eds), Elementary Differential Geometry*, Springer-Verlag.