Approach for CAD model reconstruction basing on 3D points insertion and surface approximation

Aicha Ben Makhlouf[1] , Nessrine Elloumi[2], Borhen Louhichi[3], Dominique Deneux[4]
[1]LATIS, ENISo, University of Sousse, 4023 Sousse, Tunisia
[2] SETIT, University of Sfax, 3000 Sfax, Tunisia
[3]LMS, ENISo, University of Sousse, 4023 Sousse, Tunisia
[4]CNRS, UMR 8201-LAMIH, Univ. Polytechnique Hauts-de-France, F-59313 Valenciennes, France
aychaa.benmakhlouf@gmail.com; Ellouminessrine@gmail.com; Borhen.louhichi@etsmtl.ca; Dominique.Deneux@uphf.fr

***Abstract*—Reverse engineering (RE) consists in the reconstruction of a geometric model of a 3D object from a set of points, a mesh or a 3D triangulation. This model is a combination of geometric primitives (cylinders, planes, spheres, etc.) and complex surfaces (B-Spline, NURBS…) defined by parameters that can be estimated from the 3D data. RE is widely used in different fields such as mechanic, artistic, medical, Building Information Modeling, reality augmentation, etc. In this context, the reconstruction of 3D surface is an important task to obtain the Computer Aided Design (CAD) model in order to visualize 3D objects and approximate their shapes by mathematical formulations. Triangular surface models are now commonly used to model three-dimensional object. Many of these geometric models are obtained from scanning techniques or modeled through CAD software. This paper presents a new approach to rebuild a CAD model basing on the reconstruction of the B-Spline surfaces given a set of points extracted from a deformed mesh. To guarantee a good precision of the fitted surface, new 3D points are inserted to the input mesh using the Walton's method. Given the updated set of points, the B-Spline surface is approximated. To validate the proposed method, reconstruction errors of different complex 3D surfaces before and after the points insertion are calculated. A comparison with the existing methods prove the efficiency of the developed algorithm.**

Keywords—Reverse Engineering (RE), geometric model, Computer Aided Design (CAD), 3D objects, B-Spline surface, deformed mesh, Walton's method

I. INTRODUCTION

Due to the advances in computer graphics [1], digital and 3D vision techniques [2], the three-dimensional representation of geometric data is now used in many fields: Computer Aided Design (CAD) [3], digital entertainment [4], augmented/virtual reality [5], Building Information Modeling (BIM) [6], etc. It is therefore essential to have efficient techniques for storing, exchanging and even visualizing these models. However, obtaining the most compact representation possible is not the only objective. The majority of 3D object representations are designed by CAD software. During visualization, data exchange or manufacturing processes, the geometric model must be discretized into a 3D mesh composed of a finite number of vertices and edges. Thus, in some situations the initial model may be lost or unavailable. The 3D discrete representation can also be modified, for example, after a numerical simulation, and no longer corresponds to the initial model. A reverse engineering [7][8] method is then necessary to reconstruct a continuous 3D representation from the discrete representation.

Today, the reconstruction of the CAD model from a 3D mesh [9][10] becomes an essential task in many applications (mechanic, aeronautic, medical, artistic, etc.). This model is one of the most important models used throughout the product's entire life cycle. It is used to visualize 3D objects and approximate their shapes by mathematical formulations. Adding to that, it represents the geometric support used in many other activities (analysis, manufacturing, assembly, etc.). For example, in the mechanical field, the reconstruction of the CAD model from the results of a deformed mesh (Finite Element results) facilitates the exchange between design and simulation and makes it possible to visualize and simulate the behavior of an assembly in a deformed state in order to detect possible interferences [11].

To rebuild a CAD model, it is necessary to reconstruct its different surfaces. In fact, the 3D surface reconstruction [9] is the most delicate task to obtain the CAD model.

Many researchers have proposed several algorithms for 3D surface reconstruction and CAD model rebuilding based on their types of data (point cloud of a scan, 3D mesh, Finite Element results, etc.) and fields of application (e.g., feature extraction, video games, 3D animation, CAD/CAE integration, etc.).

In this work, a new approach is proposed to reconstruct the B-Spline surfaces in order to rebuild the CAD model. To minimize the reconstruction error, the Walton's algorithm is used to insert 3D points into the initial set of points extracted from the input mesh. Then, the B-Spline surface is fitted to the updated set of points using the Levenberg Marquardt Algorithm (LMA).

This paper is organized as follows: the second section provides a literature review of 3D mesh subdivision and 3D surface reconstruction. Then, the algorithm developed to reconstruct a CAD model given a deformed mesh is presented and the original approach to reconstruct the B-spline surface by the insertion of new 3D points is described. The proposed method is validated by the evaluation of the reconstruction error and a comparison with existing methods. Conclusion is presented at the end.

2375-0138/21/$31.00 ©2021 IEEE
DOI 10.1109/IV53921.2021.00057

II. STATE OF THE ART

3D complex surfaces such as NURBS, B-Spline and Bézier are the closest surfaces to the real model of a 3D object. These surfaces are defined by a set of control points in order to interpolate/approximate the input 3D points. These points are often extracted from a triangular mesh which can be refined by inserting new points. The more these points are inserted, the more it tends to merge with control surface. The ability to insert new points and refine the 3D mesh is therefore an important problem. To refine 3D meshes many subdivision methods have been presented in the literature.

A. Subdivision methods

Subdivision methods are generally recursive algorithms. These methods start from a given mesh and then apply a proposed scheme. The process acts on the mesh by subdividing it, creating new vertices and new facets. The positions of the new points are calculated from the old points. This process produces a new mesh containing more facets than the old mesh. The new mesh can then be used as input data to the subdivision scheme, in order to be refined even more.

There are different types of subdivision schemes, namely the Catmull-Clark scheme [12], the Loop's schema [13], Kobbelt's schema [14], Doo-Sabin [15], etc.

The Catmull-Clark schema [12] is a primal schema where the initial facets are divided into several facets. This type of scheme is applied to the quadrilateral mesh. The Figure (Fig. 1) presents a refinement example according to the Catmull-Clark algorithm. The subdivision process allows the cube (Fig. 1 (a)) to get closer to the sphere (Fig. 1 (b)).

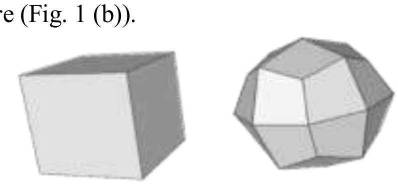

Fig. 1. Results after an iteration applied to a cube model using Catmull-Clark algorithm

Charles Loop [13] developed a subdivision method (the Loop Subdivision method) to recursively define a refined mesh by dividing each triangle into four smaller ones. Ren [16] developed another algorithm which consists in iteratively refining a given mesh to approximate the real form of the meshed object. It deals with inserting a new point between two consecutive points and each triangle is divided into four smaller triangles.

Other researchers have focused on remeshing an input 3D mesh with methods such as the Delaunay Triangulation (DT) [17], which play a vital role in mesh processing, and computation geometry. These methods are easy to implement considering the DT criterion that ensures the empty sphere/circle property, which means that the circumscribed circle/sphere of any triangle/tetrahedron in a mesh will have no node inside it.

Many researchers used the DT criterion and developed several algorithms in order to connect a set of points in space and generate a 3D mesh. Owen et al. [18] surveyed meshing techniques utilizing the Delaunay criterion and described various DT insertion methods. A typical point insertion method proposed by Weatherill and Hassan [19] aimed to generate a tetrahedral mesh scheme where nodes are inserted at a tetrahedron's centroid satisfying an underlying sizing function. Walton [20] proposed an original method to insert new points to the input mesh by the evaluation of the Bezier surface of each triangle of the mesh. This algorithm was then improved by Owen [21]. Their method attributes for each vertex of the triangulation a vector which will be used then as the normal to reconstruct the surface. From the coordinates of the three vertices of the triangle and the three calculated normal vectors on these vertices, new control points will be calculated and inserted points will be obtained.

Subdivision surfaces can be considered as techniques to approximate a given mesh to its bearing surface. However, these techniques are limited to reconstruct 3D surfaces. The subdivision surfaces can be used to add the missing information of a surface from a mesh.

B. Tridimensional surface reconstruction

The reconstruction of complex surfaces is a difficult task in reverse engineering. In fact, the reconstruction of parametric surfaces is more complicated than the reconstruction of curves since it requires a grid of control points, organized in space.

Given a triangular mesh (Fig. 2), the reconstructed surface must interpolate / approximate the points of the triangulation (a cloud of points interconnected by segments).

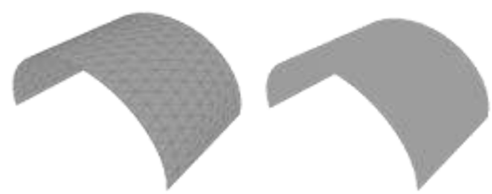

Fig. 2. Reconstruction of a complex surface from a triangular mesh

Many reconstruction methods have been developed in order to reconstruct a 3D surface from a set of points or a 3D mesh.

The Poisson method [22] is one of the strongest methods in the domain of 3D surface reconstruction which interpolates a set of oriented points by computing the gradient of an indicator function which is a vector field equal to zero almost everywhere except at points near the surface.

Volpin [23] has proposed an algorithm to reconstruct a NURBS surface from a mesh through a simplification of the input mesh. This method starts by simplifying the initial mesh model through the construction of restricted regions in curvatures according to the model. Then, a quadrilateral mesh of the model is constructed and the smooth surfaces are created using a method called the energy method.

In [24], authors proposed a method to reconstruct NURBS surfaces given unorganized 3D points without constructing patches networks or polygon meshes. Their method is based on the regularization of the point cloud and the approximation of the adequate surface. Eck and Hoppe [25] have also solved the problem of constructing NURBS surfaces from unorganized and scattered points by generating a network of B-Spline patches.

Louhichi and al. [26] developed an algorithm to reconstruct a deformed B-Spline surface given a 3D mesh. The weighted displacement estimation (WDE) method is used to solve the problem of the inconstant density of information over the deformed surface and the un-organization of points. The regular lattice of control points on the initial surface (before deformation) is calculated. To update the position of the deformed B-spline surface lattice control points, the weighted displacement of each control point is estimated. Using this lattice, the parameters of the B-spline surface are computed and the deformed reconstructed surface is obtained.

In [27], authors proposed a method to reconstruct the B-Spline surface given a set of 3D points using the Genetic Algorithm optimization approach. Another iterative algorithm based on the level set method is proposed in [28] to reconstruct the B-spline surface having as input a set of unorganized points. Moreover, the implicit B-Spline surface reconstruction method [29] is proposed to reconstruct the adequate surface from a set of points without requiring any parameterization. This method of surface reconstruction solves a system of linear equations in order to allow interaction with the user.

Ben Makhlouf et al. [10] proposed another method to reconstruct a CAD model from the deformed mesh using the Levenberg Marquardt Algorithm (LMA) to approximate B-spline surfaces. They developed another algorithm [9] basing on the Thin Plate Spline (TPS) method to optimize the locations of control points of a B-Spline surface.

The reconstruction of the tridimensional surface given a set of 3D points is a challenging task even with the development of geometric modeling systems. In this paper, a new method is proposed to reconstruct a B-Spline surface given a set of points extracted from a deformed mesh. The methodology is based on the insertion of new points using the Walton's method in order to obtain a lower reconstruction error.

III. RECONSTRUCTION OF THE CAD MODEL GIIVEN A 3D MESH

The general algorithm to reconstruct a CAD model [9][10] follows the hierarchy of the BREP model (see Fig. 3). In the first step, the 3D points corresponding to the faces and the edges are extracted from the deformed mesh. Then, new points are inserted into the initial set of points. After that, the curves (CAD edges) and the surfaces (support of faces) are reconstructed from the previous information. In the last stage, contours are added to surfaces to obtain CAD faces and the deformed model is reconstructed.

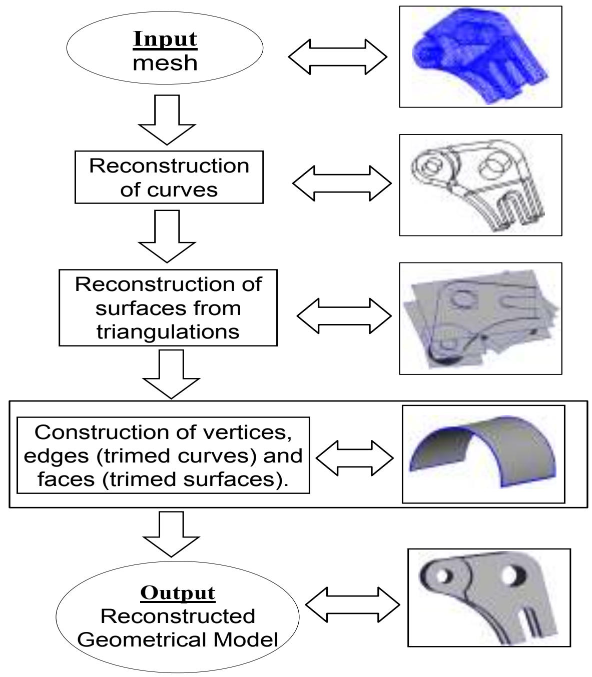

Fig.3. General algorithm to reconstruct a CAD model [9]

The main steps of the general algorithm are the following:

- Identification of the triangulation associated with each face of the model and the extraction of the set of points corresponding to the edges and the faces.
- Reconstruction of the CAD edges.
- Reconstruction of CAD faces.

For each face:

- Reconstruction of edges corresponding to the face
- Reconstruction of the bearing surface of the face. In this step, the Walton's method is adapted to insert new points and the Levenberg Marquardt Algorithm is applied to calculate the control points of the B-spline surface.
- Projection of the edges on the surface to add the loops and to obtain the face.

- Reconstruction of the deformed CAD model.

The above steps are detailed in the sections (A, B, C and D).

A. Reconstruction of the CAD edges

To reconstruct the CAD edges, the points extracted from a boundary line of a triangulated surface are interpolated. Since CAD surfaces commonly encountered are based on B-Spline surfaces, the generated edges must be compatible with their faces. Thus, B-Spline curves are used to interpolate the set of input points to obtain edges.

Given n points Q_k to interpolate with a p^{th} degree B-Spline curve, each interpolation point is described by the equation:

$$Q_k = C(\bar{u}_k) = \sum_{i=0}^{n} N_{ip}(\bar{u}_k)P_i \quad (1)$$

Where P_i are the control points of the curve, N_i are the shape functions and $\bar{u}_k$ is the parameter of the Q_k on the curve ($0 \leq \bar{u}_k \leq 1$). In order to solve the equation (1), the $\bar{u}_k$ of each Q_k is chosen on the curve and an appropriate knot vector is calculated. There are three methods for choosing the $\bar{u}_k$ parameters [30] [31]: the equally spaced method, the chord length method and the centripetal method. To reconstruct the CAD edges, the centripetal method is used because it gives the best parameterization and has the most stable behaviour when the data takes very sharp turns. This method is described by the following equation:

$$\bar{u}_0 = 0, \bar{u}_n = 0,\ \bar{u}_k = \bar{u}_{k-1} + \frac{\sqrt{|Q_k - Q_{k-1}|}}{d}, (k = 1 \dots n-1) \quad (2)$$

such as *d is* the total of square of length defined by the following equation:

$$d = \sum_{k=1}^{n} \sqrt{|Q_k - Q_{k-1}|} \quad (3)$$

In the bibliography [32], the averaging method is used to compute the knots vector from the $\bar{u}_k$ parameters. By using this method, knots reflect the distribution of the data points, which is important in order to gain accuracy in the interpolation.

$$(u_0 = \dots = u_p = 0, u_{m-p} = \dots = u_m = 1),\ u_{j+p} = \frac{1}{p}\sum_{i=j}^{j+p-1} \bar{u}_i, (j = 1 \dots n-p) \quad (4)$$

The knots vector is used to compute the B-Spline shape functions. In the last step, the control points of the curve are computed by solving Equation (4).

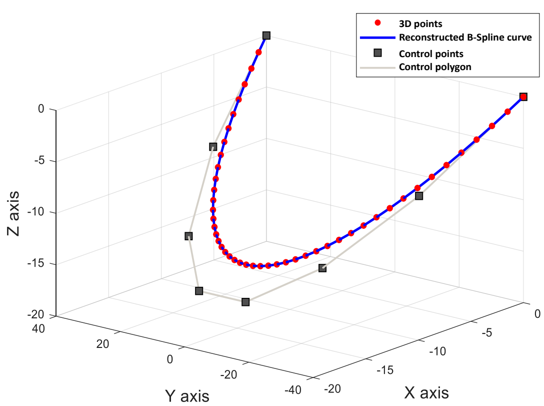

Fig. 4. Reconstruction of a B-Spline curve of degree 7

B. *Reconstruction of CAD faces*

Most of CAD systems use NURBS surfaces to model the faces of the CAD model. In this work, B-Spline surfaces are chosen (because they have the same feature as the NURBS surfaces) but induce a simpler and faster algorithm. In this regard, the reconstruction of the CAD face is based on the B-Spline surface reconstruction. To use this surface for modelling the faces, it is necessary to compute the two knots vectors in the parametric space (u,v) and to find the shape functions. Indeed, the control points should be approximated.

Figure (Fig. 5) presents the steps proposed to reconstruct the CAD face:

- Identification of the input mesh associated with each face.
- Extraction of the points corresponding to the edges.
- Reconstruction of the CAD edges.
- Extraction of the other points corresponding to the face.
- Reconstruction of the deformed surface using the proposed method.
- Generation of the 2D edges corresponding to face loops by projection of the 3D edges previously computed.
- Trim the reconstructed surface by the loop to obtain the face.

The reconstruction of the B-Spline surface is the most difficult step in this algorithm. Given the extracted set of points, each 3D surface can be interpolated / approximated. In this regard, the Walton's method, based on the points normal calculation, is used to insert new 3D points. Then, the Levenberg Marquardt Algorithm (LMA) is used to approximate the control points of the B-Spline surface based on an objective function corresponding to the sum of non-linear least squares problems [33]. The following sections describe the proposed methodology to reconstruct the B-Spline surface.

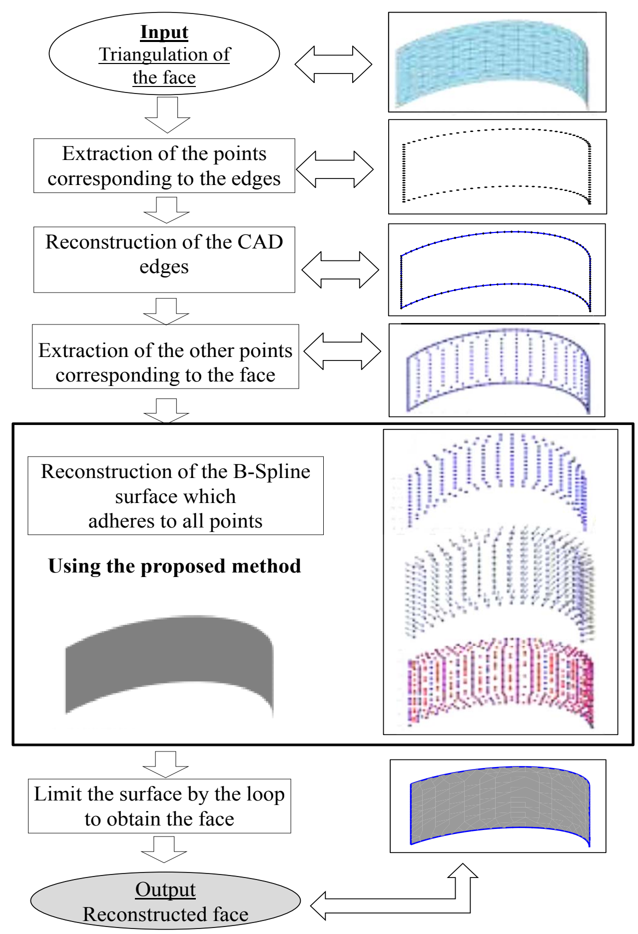

Fig. 5. Reconstruction algorithm of the CAD model faces

C. *Insertion of new points using the Walton's method*

In order to insert a set of new points to the initial input points extracted from the meshed surface, the Walton's method [20] is applied. This algorithm calculates for each node of the triangulation, a vector that represents the normal at this node to the reconstructed surface (Fig. 7).

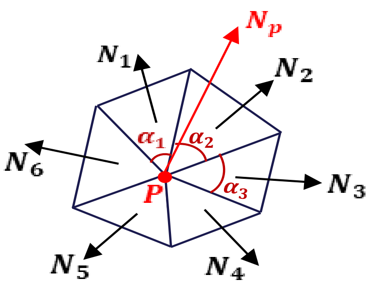

Fig. 6. Definition of the normal at a point P

The normal at a meshed surface in a node $\boldsymbol{P}$ (Fig. 6) is defined as follows (equation 5).

$$N_p = \sum_{i=1}^{n} N_i * w_i \tag{5}$$

such as:

$\boldsymbol{n}$: is the number of triangles belonging to the surface which admit the node $\boldsymbol{P}$ as a vertex.

$\boldsymbol{N_i}$: are the normal vectors at the triangles that admit the node $\boldsymbol{P}$ as a vertex.

$\boldsymbol{w_i}$: are the weighting coefficients associated with the normal vectors $\boldsymbol{N_i}$, defined by the equation (6).

$$w_i = \frac{\alpha_i}{\sum_{i=1}^{n} \alpha_i} \tag{6}$$

Given a triangle defined by three points $\boldsymbol{P_0}$, $\boldsymbol{P_1}$ and $\boldsymbol{P_2}$ and three vectors $\boldsymbol{N_0}$, $\boldsymbol{N_1}$ et $\boldsymbol{N_2}$, this method evaluates the portion of the surface that will be reconstructed and that corresponds to this triangle (Fig. 9).

The equation to calculate an inserted point in the triangular surface is described by the equation (7).

$$S(u,v,w) = \sum_{i+j+k=4} P_{i,j,k} \frac{4!}{i!\,j!\,k!} u^i v^j w^k, \tag{7}$$

$$u, v, w \geq 0; u + v + w = 1; i, j, k \geq 0$$

such as $\boldsymbol{u}$, $\boldsymbol{v}$ and $\boldsymbol{w}$: are the coordinates of a point in the triangle (Fig. 8).

$\boldsymbol{P_{i,j,k}}$: Are the control points (Fig. 8) of this portion of the surface (Bezier surface calculated from a triangle), depending on $\boldsymbol{P_0}$, $\boldsymbol{P_1}$, $\boldsymbol{P_2}$, $\boldsymbol{N_0}$, $\boldsymbol{N_1}$ and $\boldsymbol{N_2}$.

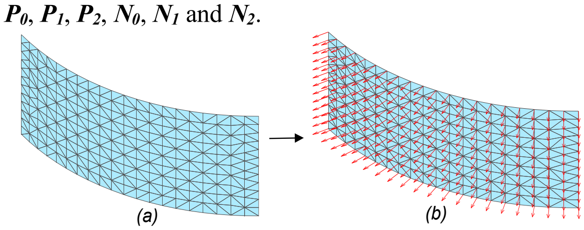

Fig. 7. Calculated normal for each node of the triangulation

(a) *The input mesh (b) Calculated normal at each point*

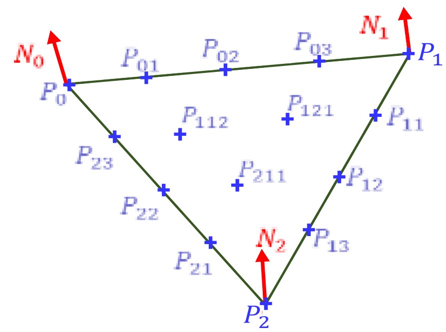

Fig. 8. Control points of the triangle

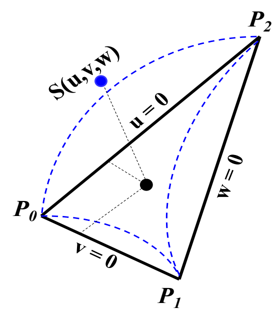

Fig. 9. *Coordinates of a point in a triangle*

In order to insert a new point on the surface using the Walton's method, three-control point P_{ij} should be calculated for each triangle edge.

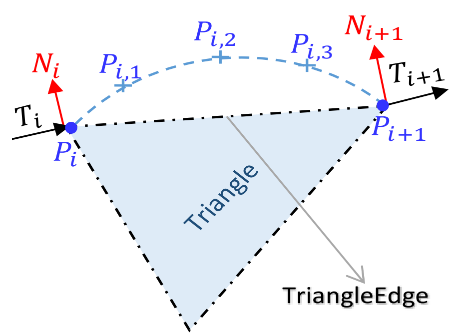

Fig. 10. Control points on TriangleEdge

As shown in (Fig. 10), the selected edge has two end points P_i and P_{i+1} and two tangents T_i and T_{i+1}

Thus, the tangents are calculated using equation (8):

$$T_i = T_{i+1} = \frac{P_{i+1} - P_i}{||P_{i+1} - P_i||} \tag{8}$$

The control points of each TriangleEdge$_{\mathbf{(i)}}$ (Fig. 10) are calculated using equations (9) to (18):

$$d_i = ||P_{i+1} - P_i|| \quad (9)$$

$$a_i = N_i . N_{i+1} \quad (10)$$

$$a_{i,0} = N_i . T_i \quad (11)$$

$$a_{i,1} = N_{i+1} . T_{i+1} \quad (12)$$

$$\rho_i = \frac{6(2a_{i,0} + a_i . a_{i,1})}{4 - a_i^2} \quad (13)$$

$$V_{i,1} = P_i + \frac{d_i(6T_i - 2\rho N_i + \sigma N_{i+1})}{18} \quad (14)$$

$$V_{i,2} = P_{i+1} - \frac{d_i(6T_{i+1} + \rho N_i + 2\sigma N_{i+1})}{18} \quad (15)$$

$$P_{i,1} = \frac{1}{4} P_i + \frac{3}{4} V_{i,1} \quad (16)$$

$$P_{i,2} = \frac{1}{2} V_{i,1} + \frac{1}{2} V_{i,2} \quad (17)$$

$$P_{i,3} = \frac{3}{4} V_{i,2} + \frac{1}{4} P_{i+1} \quad (18)$$

Using the control points on the TriangleEdges, The control points $P_{1,1,2}$, $P_{1,2,1}$ and $P_{2,1,1}$ within the triangle are defined (Fig. 7). The locations of these points are essential to describe the continuity between adjacent triangles.

Walton [20] proposed a method in which the inner control points are defined for each TriangleEdge, resulting in six vectors $G_{i,j}$ {i=0,1,2; j=0,1}.

The final locations of the inner control points $P_{1,1,2}$, $P_{1,2,1}$ and $P_{2,1,1}$ are evaluated as a function of the parameters (u, v, w) and $G_{i,j}$.

For each TriangleEdge$_{(i)}$ in the triangle, the following parameters are calculated:

$$W_{i,0} = V_{i,1} - P_i \quad (19)$$

$$W_{i,1} = V_{i,2} - V_{i,1} \quad (20)$$

$$W_{i,2} = P_{i+1} - V_{i,2} \quad (21)$$

$$D_{i,0} = P_{i,3} - \frac{1}{2}(Pi,1 + P_i) D_{i,1} = P_{i,1} - \frac{1}{2}(Pi,1 + P_{i,3}) \quad (22)$$

$$A_{i,0} = \frac{N_i \times W_{i,0}}{||W_{i,0}||} \quad (23)$$

$$A_{i,2} = \frac{N_{i+1} \times W_{i,2}}{||W_{i,2}||} \quad (24)$$

$$A_{i,1} = \frac{A_{i,0} + A_{i,2}}{||A_{i,0} + A_{i,2}||} \quad (25)$$

$$\lambda_{i,0} = \frac{D_{i,0} . W_{i,0}}{W_{i,0} . W_{i,0}} \quad (26)$$

$$\lambda_{i,1} = \frac{D_{i,1} . W_{i,2}}{W_{i,2} . W_{i,2}} \quad (27)$$

$$\mu_{i,0} = D_{i,0} . A_{i,0} \quad (28)$$

$$\mu_{i,1} = D_{i,1} . A_{i,2} \quad (29)$$

The two control points $G_{i,0}$, $G_{i,1}$ with respect to TriangleEdge$_{(i)}$ is defined as:

$$G_{i,0} = \frac{1}{2}(P_{i,1} + P_{i,2}) + \frac{2}{3}\lambda_{i,0} W_{i,1} + \frac{1}{3}\lambda_{i,1} W_{i,0} + \frac{2}{3}\mu_{i,0} A_{i,1} + \frac{1}{3}\mu_{i,1} A_{i,0} \quad (30)$$

$$G_{i,1} = \frac{1}{2}(P_{i,2} + P_{i,3}) + \frac{1}{3}\lambda_{i,0} W_{i,1} + \frac{2}{3}\lambda_{i,1} W_{i,0} + \frac{1}{3}\mu_{i,0} A_{i,1} + \frac{2}{3}\mu_{i,1} A_{i,0} \quad (31)$$

Interior control points $P_{1,1,2}$, $P_{1,2,1}$ and $P_{2,1,1}$ may be uniquely evaluated as:

$$P_{1,1,2} = \frac{1}{u+v}(uG_{2,2} + vG_{0,1}) \quad (32)$$

$$P_{1,2,1} = \frac{1}{w+u}(wG_{0,2} + uG_{1,1}) \quad (33)$$

$$P_{2,1,1} = \frac{1}{v+w}(vG_{1,2} + wG_{2,1}) \quad (34)$$

Finally, to evaluate the inserted point at parameter (u,v,w), equation (7) may be employed by using the control points defined in equations (16) to (18) and (32) to (34).

This process is executed for each point extracted from the input triangulation. Then, given the updated set of points, the B-Spline surface should be reconstructed.

D. Reconstruction of the B-Spline surface using LMA

The B-Spline surface [34] is defined with its degree, the values of the knots vectors and its control points. This surface is described on each point using equation (35):

$$S(u,v) = \sum_{i=0}^{n} \sum_{j=0}^{m} N_{i,p}(u) N_{j,q}(v) P_{i,j} \quad (35)$$

such as u and v are the knots vectors which have respectively $(r+1)$ and $(s+1)$ nodes in the interval [0,1], where $r = n + p + 1$ and $s = m + q + 1$. This surface is defined by the degree p in the $u-$direction and the degree q in the $v-$direction (u and v are the parameters that define each point on the surface).

Indeed, $N_{i,p}$ and $N_{j,q}$ are the B-Spline basis functions of degree p in the u-direction and degree q in the v-direction respectively. The above basis functions are calculated in a recursive way:

$$N_{i,0}(u) = \begin{cases} 1 \; if \; u_i \leq \mathrm{u} \leq u_{i+1} \\ 0 \; otherwise \end{cases} \quad (36)$$

$$N_{i,p}(u) = \frac{u - u_i}{u_{i+p} - u_i} . N_{i,p-1}(u) + \frac{u_{i+p+1} - u}{u_{i+p+1} - u_{i+1}} . N_{i+p+1,p-1}(u) \quad (37)$$

In order to reconstruct the 3D surface, the reconstruction algorithm follows the following steps:

(1) Calculate the infinite average plane given the updated 3D points (the initial points and the inserted ones).
(2) Project the points on the average plane
(3) Find the bounding box of the projected points
(4) Convert the projected 3D points to (u, v) coordinates
(5) Optimize the control points using the Levenberg Marquardt Algorithm (LMA).
The most difficult step is to find the control points of the B-Spline surface. Thus, the LMA is applied in order to optimize the locations of these control points.

Optimization of the control points of the B-Spline surface using LMA

Given L points $\boldsymbol{q}_l$ of parameter $s(u_l, v_l)$, the main idea of this algorithm is to iteratively find the vector $\boldsymbol{p}$ relating to the control points that minimizes the objective function $J(p)$.

$$J(p) = \sum d_i^2(p) \tag{38}$$

Where: d_i is the distance from the points q_l to the B-Spline surface defined by the vector p containing the control points found on each iteration.

By applying the Levenberg Marquardt algorithm to reconstruct the B-Spline surface, the vector of control points is calculated by minimizing the following objective function:

$$J(\boldsymbol{P}) = \sum_{l=0}^{L-1} \left(\sum_{k=0}^{K-1} \boldsymbol{p}_k B_k(u_l, v_l) - \boldsymbol{q}_l \right)^2 \rightarrow \min \tag{39}$$

such as $K = (N+1)\times(M+1)$ is the number of control points and $B_{j\times(N+1)+i} = N_i(u)N_j(v)$ is a matrix of size $L\times K$ defined as following:

$$\begin{pmatrix} B_0(u_0,v_0) & B_1(u_0,v_0) & \cdots & B_{K-1}(u_0,v_0) \\ B_0(u_1,v_1) & B_1(u_1,v_1) & \ldots & B_{K-1}(u_1,v_1) \\ \vdots & \vdots & \vdots & \vdots \\ B_0(u_{L-1},v_{L-1}) & B_1(u_{L-1},v_{L-1}) & \ldots & B_{K-1}(u_{L-1},v_{L-1}) \end{pmatrix} \tag{40}$$

IV. VALIDATION

To validate the accuracy of the proposed approach, reconstruction errors before and after the insertion of points of three different B-Spline surfaces are calculated using two metrics: the average distance and the Maximum Root Mean Square Error (MRMS).
The average distance [10] calculates the error of the reconstructed surface by simply computing the mean of distances between the points of the initial surface and the points of the reconstructed surface.
The MRMS [35] is a simple metric to measure the distance between two surfaces in 3D space using the following equation:

$$d(p, S') = \min_{p' \in S'} \left\| p - p' \right\|_2 \tag{41}$$

Where d(p, S') is the distance between a point p in the surface S and a point p' in the surface S'.
$\left\| p - p' \right\|_2$ represents the Euclidean distance between the two points p and p'.
Thus, the root mean square error between the two surfaces S and S' (d_{RMS}) is defined by the equation (42).

$$d_{RMS}(S,S') = \sqrt{\frac{1}{|S|} \iint_{P\in S} d(p,S')^2 dS} \tag{42}$$

Where, |S| represents the surface S.
The MRMS distance is calculated using equation (43).

$$MRMS = \max(d_{RMS}(S,S'), d_{RMS}(S',S)) \tag{43}$$

The first example (Fig. 11) presents the reconstruction of a B-Spline surface selected from a deformed CAD model. The number of input points (extracted from the meshed surface) is 237. After the insertion process, the number of points to fit becomes 645 (Number of facets of the input mesh = 408).

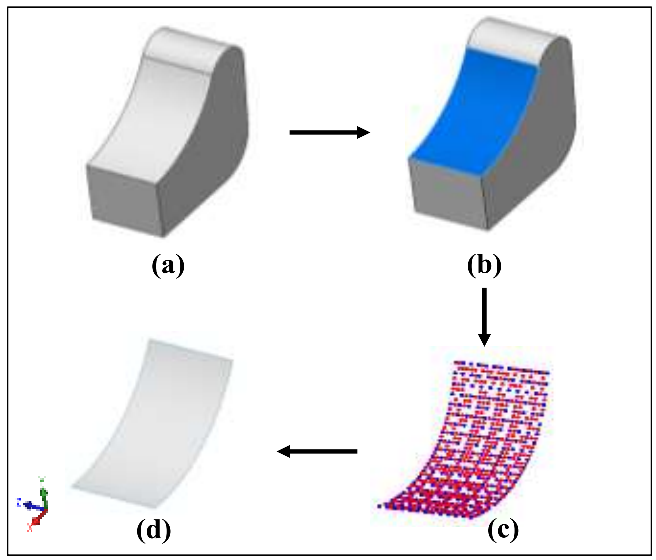

Fig. 11. Reconstruction of the first B-Spline surface (a)CAD model (b) the selected face (c) the extracted points (blue points) and the inserted points (red points), (d) the reconstructed B-Spline surface

The errors (in mm) of the first surface reconstruction given the initial points and the updated set of points (the initial and the inserted points) are presented in (Table 1).

Table 1: Reconstruction results of the reconstructed surface before and after the insertion of points.

	Number of points	Error using the average distance	Error using MRMS
Reconstructed surface using **LMA [10] without points insertion**	237	2,003 E-06	0,0018
Reconstructed surface after insertion of points using the **Loop Subdivision method [13]**	881	2,169 E-05	0,0187
Reconstructed surface after insertion of points using **the proposed approach**	645	4,269 E-07	6,700 E-04

Figure (Fig. 12) represents a second example of another B-Spline surface reconstruction.

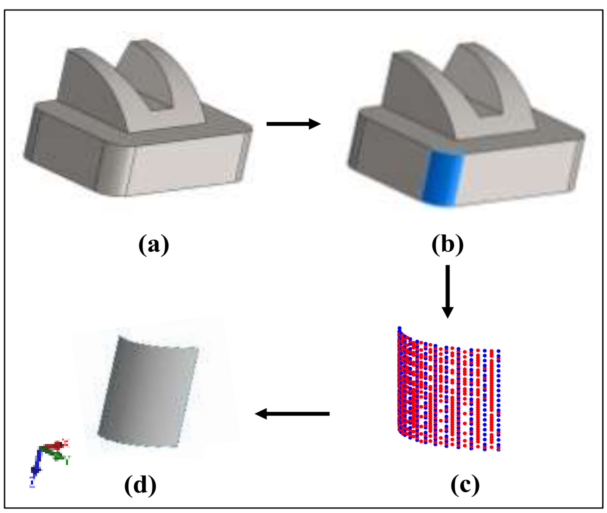

Fig. 12. Reconstruction of the second B-Spline surface (a) CAD model (b) the selected face (c) the extracted points (blue points) and the inserted points (red points), (d) the reconstructed B-Spline surface

The errors (in mm) of the second surface reconstruction given the initial points and the updated set of points (the initial and the inserted points) are presented in (Table 2).

Table 2: Reconstruction results of the second reconstructed surface before and after the insertion of points.

	Number of points	Error using the average distance	Error using MRMS
Reconstructed surface using **LMA [10] without points insertion**	273	6,364 E-06	0,0051
Reconstructed surface after insertion of points using the **Loop Subdivision method [13]**	1025	3,633 E-05	0,0277
Reconstructed surface after insertion of points using **the proposed approach**	753	5,063 E-07	0,0011

Figure (Fig. 13) represents a third example of the B-Spline surface reconstruction.

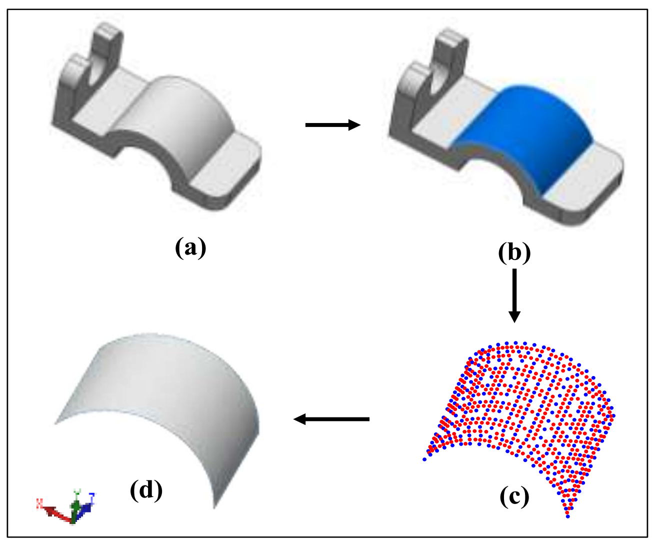

Fig. 13. Reconstruction of the third B-Spline surface (a) CAD model (b) the selected face (c) the extracted points (blue points) and the inserted points (red points), (d) the reconstructed B-Spline surface

The errors (in mm) of the third surface reconstruction given the initial points and the updated set of points (the initial and the inserted points) are presented in (Table 1).

Table 3: Reconstruction results of the third reconstructed surface before and after the insertion of points.

	Number of points	Error using the average distance	Error using MRMS
Reconstructed surface using **LMA [10] without points insertion**	233	1,339 E-04	0,1385
Reconstructed surface after insertion of points using the **Loop Subdivision method [13]**	865	5,512 E-04	0,6499
Reconstructed surface after insertion of points using **the proposed approach**	633	1,155 E-05	0,0114

The results shown in (Table 1), (Table 2) and (Table 3) demonstrate that the error of the surface reconstruction using LMA [10] given the initial set of points and the error of the surface reconstruction given the updated set of points using the Loop subdivision method [13] are higher than the reconstruction errors of the same surfaces with inserted points using the proposed approach applying the two quality metrics (average distance [10] and MRMS [35]).

As result, the proposed approach, based on the insertion of the new points using the Walton's method improves the reconstruction quality and guarantees more precision of the reconstructed surfaces. Although the proposed method is limited to reconstruct CAD models from only triangular meshes, it is efficient and guarantees better results.

V. CONCLUSION

In this paper, an original method is proposed to reconstruct a CAD model by the approximation of B-Spline surfaces. A methodology for the approximation of B-Spline surface given an input mesh has been presented. In the first part, previous works are briefly introduced. In the second part, the general algorithm to reconstruct a CAD model, is described. Then, the proposed approach to reconstruct the 3D surface is detailed; The Walton's method is adapted in order to add new points to the set of points extracted from the mesh. Then, given the updated set of points, the control points of the B-Spline surface are optimized using the Levenberg Marquardt Algorithm. Finally, validation examples present the results obtained by the reconstruction of different 3D surfaces extracted from three CAD models. Moreover, a comparison with existing methods proves that the proposed approach is more precise giving a good quality of reconstruction.

REFERENCES

[1] J. Posada, M. Zorrilla, A. Dominguez, B. Simoes, P. Eisert, D. Stricker, J. Ram-bach, J. Döllner, and M.Guevara, "Graphics and media technologies for operators in industry 4.0," IEEE computer graphics and applications, vol. 38, no. 5, pp. 119–132, 2018.

[2] V. Wiley and T. Lucas, "Computer vision and image processing: a paper review," International Journal of Artificial Intelligence Research, vol. 2, no. 1, pp. 29–36, 2018.

[3] T. W. Sederberg. "Computer Aided Geometric Design". Computer Aided Geometric Design Course Notes. January 10, 2012.

[4] V. Drosos, C. Alexakos, A. Alexandri, and D. Tsolis, "Evaluating 3D Serious Games on Cultural Education," in 2018 9th International Con-ference on Information, Intelligence, Systemsand Applications (IISA). IEEE, 2018, pp. 1–5.

[5] M. V. Voronina, Z. O. Tretyakova, E.G.Krivonozhkina, S. I. Buslaev, and G.G.Sidorenko, "Augmented Reality in Teaching Descriptive Geometry, Engineering and Computer Graphics--Systematic Review and Results of the Russian Teachers' Experience." EURASIA Journal of Mathematics, Science and Technology Education, vol. 15, no. 12, 2019.

[6] A. Hamieh, A. B. Makhlouf, B. Louhichi, and D. Deneux, "A BIM-based method to plan indoor paths," Automation in Construction, vol. 113, p.103120, 2020.

[7] R. Bénière, G. Subsol, G. Gesquière, F. Le Breton, and W. Puech, "A comprehensive process of reverse engineering from 3D meshes to CAD models," Computer-Aided Design, vol. 45, no. 11, pp.1382–1393, 2013.

[8] A. B. Makhlouf, B. Louhichi, M. A. Mahjoub, and G. Subsol, "Approach for CAD model reconstruction from a deformed mesh," in 2017 IEEE/ACS 14th International Conference on Computer Systems and Applications (AICCSA). IEEE, 2017, pp. 327–333.

[9] A. Makhlouf, B. Louhichi, D. Deneux, and M. A. Mahjoub, "Reconstruction of the CAD model using TPS surface", in 2019 23rd International Conference Information Visualisation (IV). IEEE, 2019, pp. 417– 424.

[10] A. Ben Makhlouf, B. Louhichi, M. A. Mahjoub,and D. Deneux, "Reconstruction of a CAD model from the deformed mesh using B-spline surfaces," International Journal of Computer Integrated Manufacturing, vol. 32, no. 7, pp. 669–681, 2019.

[11] B. Louhichi, M. Tlija, A. Ben amara, and A. Tahan, "An algorithm for CAD tolerancing integration: generation of assembly configurations according to dimensional and geometrical tolerances," Computer-Aided Design, vol. 62, pp.259–274, 2015.

[12] K. Bandara and F. Cirak, "Isogeometric shape optimisation of shell structures using multiresolution subdivision surfaces", Computer-Aided Design, vol. 95, pp. 62–71, 2018.

[13] G. Li, C. Ren, J. Zhang, and W. Ma, "Approximation of Loop subdivision surfaces for fast rendering," IEEE transactions on visualization and computer graphics, vol. 17, no. 4, pp. 500–514, 2010.

[14] L. P. Kobbelt, "Discrete fairing and variational subdivision for freeform surface design," The Visual Computer, vol. 16, no. 3, pp. 142–158, 2000.

[15] H. S. Bølviken, J. Bersvendsen, F. Orderud,S. R. Snare, P. Brekke, and E. Samset, "Two methods for modifed Doo–sabin modeling of non-smooth surfaces—applied to right ventricle modeling," Journal of Medical Imaging, vol. 7, no. 6,p. 067001, 2020.

[16] B. Ren and I. Hagiwara, "Composite freeform surface reconstruction using recursive interpolating subdivision scheme," Computers in Indus-try, vol. 50, no. 3, pp. 265–275, 2003.

[17] B. Delaunay et al., " A la mémoire de Georges Voronoï. Izv. Akad. Nauk SSSR," Otdelenie Matematicheskih i Estestvennyh Nauk, vol. 7, pp. 793–800, 1934.

[18] S. J. Owen, "A survey of unstructured mesh generation technology." IMR, vol. 239, p. 267, 1998.

[19] N. P. Weatherill and O. Hassan, "Efficient three-dimensional Delaunay triangulation with automatic point creation and imposed boundary con-straints," International journal for numerical methods in engineering, vol. 37, no. 12, pp. 2005–2039, 1994.

[20] D. Walton and D. Meek, "A triangular G1 patch from boundary curves," Computer-Aided Design, vol. 28, no. 2, pp. 113-123, 1996.

[21] S. J. Owen and D. R. White, “Mesh-based geometry”, International Journal for Numerical Methods in Engineering, vol. 58, no. 2, pp. 375–395, 2003.

[22] M. Kazhdan, M. Bolitho, and H. Hoppe, “Poisson surface reconstruction,” in Proceedings of the Fourth Eurographics Symposium on Geometry Processing, Aire-la-Ville, Switzerland, Switzerland, vol 7, 2006, pp. 61–70.

[23] O. Volpin, A Sheffer, M Bercovier and L. Joskowicz, “Mesh simplification with smooth surface reconstruction”, Computer-aided Design, vol. 30, no. 11, pp. 875–882, 1998.

[24] N. Leal, E. Leal, and J. W. Branch, “Simple method for constructing nurbs surfaces from unorganized points”, in Proceedings of the 19th International Meshing Roundtable, Springer, Berlin, Heidelberg, 2010, pp. 161–175.

[25] M. Eck and H. Hoppe, “Automatic reconstruction of B-spline surfaces of arbitrary topological type”, in Proceedings of the 23rd annual conference on Computer graphics and interactive techniques, 1996, pp. 325–334.

[26] B. Louhichi, G. N. Abenhaim, and A. S. Tahan, “CAD/CAE integration: updating the CAD model after a FEM analysis,” The International Journal of Advanced Manufacturing Technology, vol. 76,no. 1-4, pp. 391–400, 2015.

[27] C. D. Chandra and A. P. Kumar, “B-Spline Surface Reconstruction by Control Point Optimization Using Genetic Algorithm,” International Journal of Mechanical and Industrial Technology (IJMIT), vol. 1, no. 1, pp. 31–39, 2014.

[28] P. Wen, X. Wu, Y. Zhu, and X. Peng, “LS-RBF network based 3D surface reconstruction method,” in 2009 Chinese Control and Decision Conference. IEEE, 2009, pp. 5785–5789.

[29] M. Rouhani, A. D. Sappa, and E. Boyer, “Implicit B-spline surface reconstruction,” IEEE Transactions on Image Processing, vol. 24, no. 1, pp. 22–32, 2014.

[30] C. Manni, A. Reali, and H. Speleers, “Isogeometric collocation methods with generalized B-splines,” Computers & Mathematics with Applications, vol. 70, no. 7, pp. 1659–1675, 2015.

[31] P. Gunz and P. Mitteroecker, “Semilandmarks: a method for quantifying curves and surfaces,” Hystrix the Italian journal of mammalogy, vol. 24, no. 1, pp. 103–109, 2013

[32] M. Amirfakhrian, “Approximation of 3D-Parametric functions by bicubic B-spline functions”, International Journal of Mathematical Modelling & Computations, vol. 2, no. 3 (SUM-MER), pp. 211–220, 2016.

[33] C. M. Shakarji, “Least-squares fitting algorithms of the NIST algorithm testing system,” Journal of research of the National Institute of Standards and Technology, vol. 103, no. 6, p. 633, 1998

[34] L. Piegl and W. Tiller, The NURBS book. Springer Science & Business Media, 1996.

[35] Y. Niu, Y. Zhong, W. Guo, Y. Shi, and P. Chen, “2D and 3D image quality assessment: A survey of metrics and challenges,” IEEE Access, vol. 7, pp. 782–801, 2018.

[36] N. Elloumi, H. L. H. Kacem, N. Dey, A. S. Ashour, and M. S. Bouhlel, “Perceptual metrics quality: Comparative study for 3D static meshes,” International Journal of Service Science, Management, Engineering, and Technology (IJSSMET), vol. 8, no. 1, pp. 63–80, 2017.