

## 2D IMAGE TO STANDARD TRIANGLE LANGUAGE (STL) 3D IMAGE CONVERSION

<sup>1</sup>Dr.K.Spurthi, Associate Professor , CSE(AIML), Kolluspoorthy03@gmail.com  
Swarna Bharathi institute of science and technology,  
Khammam,

<sup>2</sup>B.Yugandhara Chary ,Assistant Professor, CSE(AIML), yugandhar.bandla@gmail.com  
Swarna Bharathi institute of science and technology,  
Khammam

<sup>3</sup>Ameena nasreen, Assistant Professor, CSE(AIML), amena.nasreen.md@gmail.com  
Swarna Bharathi institute of science and technology,  
Khammam

### ABSTRACT:

STereoLithography, often known as Standard Tessellation Language (STL), is a 3D printing format that has seen tremendous growth in popularity recently. The STL file format was developed by 3D systems. By utilizing triangles, the STL file format is able to approximate the surfaces of a solid object even in the absence of color and texture. There are two formats that are specified: binary and ASCII. In this work, we offer a method that can take a 2D image and turn it into a 3D model using an STL file.  
Keywords STereoLithography · STL · 2D image · 3D image conversion · Triangulation · Mesh

### INTRODUCTION

These days, 3D printing is all the rage, with many uses in fields as diverse as medicine, manufacturing, art, design, and medical equipment. [1] It defines two different types of formats: binary and ASCII. 3D printing, stereolithography, computer-aided design and manufacturing (CAD & CAM), and laminating object manufacturing algorithms are the main methods for creating models of machinery in three dimensions [2]. The geometrical representation of an actual three-dimensional object is a mesh of triangles in the STL file format, which stands for the image surface represented by three ordered coordinate points. It explains a unit normal, amorphous triangulated surface with triangles as its nodes and a

Cartesian coordinate system that includes negative and positive coordinates. The STL image format allows for arbitrary units and does not include scaler data [3]. Despite STL files' many useful applications in 3D printing, they aren't without their limitations. For one, they only capture the model's external shape and surface. Secondly, drawing the model's overlaps and gaps can be challenging due to the non-manifold edges. Lastly, STL files don't include the model's texture, color, attributes, or internal structure.

### STL File Structure

Figure 1 shows that STL files are surface models for three-dimensional objects. The surface space is surrounded by tiny triangles, and its records include the vertex coordinates and the triangle normal vector [4]. In order to prevent these mistakes, the STL file model takes into account the following conditions:  
Right Hand Rule: To comprehend the orientation of the axes, one uses the right hand rule. Inside the three-dimensional space, axis coordination between the right thumb and the Z-axis is maintained by X-axis movement to the Y-axis and right-handed coordinates in a positive direction as shown by the figure curl. Turn the tables on handedness by swapping out gives the same result when the axis directions are inverted, and labels the axes [5]

The Share Vertex Rule states that adjacent triangles can only have two vertices linked with them.

Two triangles can share a single edge line.

Adjacent Information: A triangle is made up of three interconnected triangles.

As mentioned earlier, two formats of Sfile are: (1) binary structure (2) ASCII structure [6].

### STL-ASCII Format

STL-ASCII is a basic text format that indicates the triangle mesh. of the surface of an object [7].

Nevertheless, the file size is quite enormous, which is the only reason it is rather useless. Fig. provides the syntax for the ASCII STL file. 2.

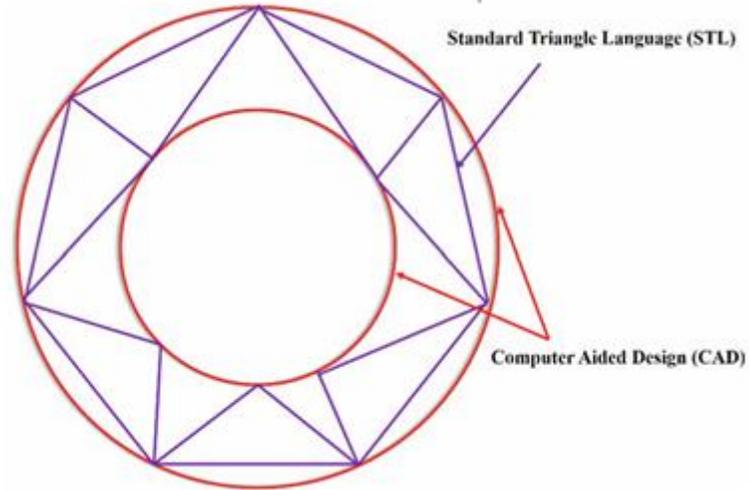


Fig. 1 CADrepresentation and STL approximation (3D Systems, 1987)

```
solid_object
  primary_loop ni nj nk
    secondary_loop
      grid v1x v1y v1z
      grid v2x v2y v2z
      grid v3x v3y v3z
    end secondary_loop
  end primary_loop
end solid_object
```

Fig.2 SyntaxofSTL-ASCIIfile[7]

### 1.1 STL Binary Format

STL Binary uses IEEE floating point and integer representation[8].The syntax of STL Binary file is given in Fig.3.

| Bytes  | Data type             | Description                   |
|--|-----------------------|-------------------------------|
| 80   | ASCII                 | Header. No data significance. |
| 4  | unsigned long integer | Number of facets in file      |
| <div style="display: flex; align-items: center; justify-content: center;"> <div style="font-size: 4em; margin-right: 10px;">}</div> <div style="display: flex; flex-direction: column; gap: 5px;"> <div style="display: flex; align-items: center;"> <div style="font-size: 2em; margin-right: 5px;">4</div> <div style="margin-right: 5px;">float</div> <div style="margin-right: 5px;">i for normal</div> </div> <div style="display: flex; align-items: center;"> <div style="font-size: 2em; margin-right: 5px;">4</div> <div style="margin-right: 5px;">float</div> <div style="margin-right: 5px;">j</div> </div> <div style="display: flex; align-items: center;"> <div style="font-size: 2em; margin-right: 5px;">4</div> <div style="margin-right: 5px;">float</div> <div style="margin-right: 5px;">k</div> </div> <div style="display: flex; align-items: center;"> <div style="font-size: 2em; margin-right: 5px;">4</div> <div style="margin-right: 5px;">float</div> <div style="margin-right: 5px;">x for vertex 1</div> </div> <div style="display: flex; align-items: center;"> <div style="font-size: 2em; margin-right: 5px;">4</div> <div style="margin-right: 5px;">float</div> <div style="margin-right: 5px;">y</div> </div> <div style="display: flex; align-items: center;"> <div style="font-size: 2em; margin-right: 5px;">4</div> <div style="margin-right: 5px;">float</div> <div style="margin-right: 5px;">z</div> </div> <div style="display: flex; align-items: center;"> <div style="font-size: 2em; margin-right: 5px;">4</div> <div style="margin-right: 5px;">float</div> <div style="margin-right: 5px;">x for vertex 2</div> </div> <div style="display: flex; align-items: center;"> <div style="font-size: 2em; margin-right: 5px;">4</div> <div style="margin-right: 5px;">float</div> <div style="margin-right: 5px;">y</div> </div> <div style="display: flex; align-items: center;"> <div style="font-size: 2em; margin-right: 5px;">4</div> <div style="margin-right: 5px;">float</div> <div style="margin-right: 5px;">z</div> </div> <div style="display: flex; align-items: center;"> <div style="font-size: 2em; margin-right: 5px;">4</div> <div style="margin-right: 5px;">float</div> <div style="margin-right: 5px;">x for vertex 3</div> </div> <div style="display: flex; align-items: center;"> <div style="font-size: 2em; margin-right: 5px;">4</div> <div style="margin-right: 5px;">float</div> <div style="margin-right: 5px;">y</div> </div> <div style="display: flex; align-items: center;"> <div style="font-size: 2em; margin-right: 5px;">4</div> <div style="margin-right: 5px;">float</div> <div style="margin-right: 5px;">z</div> </div> </div> </div> <div style="font-size: 4em; margin-left: 10px;">}</div> <div style="margin-left: 10px;">+</div> |                       |                               |

Fig.3 SyntaxofSTLbinaryfile[8]

## STL FILE SURFACE REPRESENTATION

A tessellation pipeline is used to build a triangular mesh for the 3D object surface of the boundary [3]. However, mesh also have the data to formulate the

side of triangle which contain interior of 3D object. The STL file format is triangular representation in which the triangles are represented by the vertices and are subsequently used for 3D model construction. Such information is about the triangles orientation and can be represented with numerical values.

$$\begin{matrix}
 n_x^i & n_y^i & n_z^i \\
 v_{1x}^i & v_{1y}^i & v_{1z}^i \\
 v_{2x}^i & v_{2y}^i & v_{2z}^i \\
 v_{3x}^i & v_{3y}^i & v_{3z}^i
 \end{matrix}$$

An outward normal coordinate is represented by  $n_l$   $x_l, n_l y_l, n_l z_l$  for the  $l$ th facet and  $j$ th vertex coordinates from the  $l$ th triangle are  $v_l x_l, n_l y_l, n_l z_l$  where  $l = 1, 2, \dots, N$ .

### STL MESH CONSTRUCTION

In three-dimensional modeling, polygonal mesh models with triangulating points are commonly used. Rendering and design make extensive use of these models. 3D modeling using STL files The amount and quality of polygons determine the image quality [9]. Different Talton [10] and Cignoni et al. surveyed methods for simplifying meshes. Luebke [12] and [11]. A mesh simplification approaches can be categorized in three general groups such as.

### VERTEX CLUSTERING

In vertex clustering [13, 16], the idea was to map all of the points to the same pixel such that only one point could be seen, with the rest buried by hidden surface removal. This causes it to squander  $n$  points while rendering, and we end up with an unsatisfactory image. To reduce this wastage, pick out points that land on the same pixel and use them to generate a new point for representation and display. Similar principles underpin the vertex clustering algorithm's operation. In an effort to discover nearby vertices, it creates a new vertex that looks like a point and attempts to represent them.

Assume that the graph  $G = (V, E)$  has vertices ("V") and edges ("E"). The vertex's  $v_k$  and  $v_l$  are linked by an edge, which is denoted as  $e_{kl}$ . The neighbor who is linked is

$$N_k = \{v_l : e_{kl} \in E, e_{jk} \in E\}$$

### EDGE CONTRACTION

In order to understand edge contraction, one must first understand the idea of vertex edge collapse [12, 13]. Simplification algorithms often use edge

contraction. In this idea, the edges of two vertices at opposite ends are combined to form a single edge. At these two points, faces that were incident on the edge melded into a single new edge, sparing the surrounding neighborhood.

### VERTEX DECIMATION

The objective of vertex dissemination [12–15] is to minimize the triangles in mesh without affecting the original topology and geometry. For Average plane vertex  $v_l$  normal of neighboring triangles  $n_l$  centers  $c_l$  and area is  $A_l$ . The weighted average is

$$n = \frac{\sum_i A_i n_i}{\sum_i A_i} \quad (2)$$

$$\hat{n} = \frac{n}{|n|} \quad (3)$$

$$n = \frac{\sum_i A_i y_i}{\sum_i A_i} \quad (4)$$

Distance is

$$|\hat{n} = (v_j - y)$$

## 2. EXPERIMENTAL PROCESS

A wide variety of industries, including healthcare, industry, the arts, and design, are making use of 3D printing, which has seen a meteoric rise in popularity. [1] There is a need for 2D contours with triangle matrices in STL files. In response to z-values, STL files generate a succession of polygons. It is necessary to link the STL image surface. In Figure 4, we can see the proposed algorithm in all its complexity. We have taken one input image in the experimental setup, as shown in Fig. 5. The following step is to determine the color detection tolerance value and the image's form width; in this case, 156 pixels.

Images with permissible subtle variations in color tone can benefit from color detection tolerance. It has been configured to different

color tolerance for this image. Figure 6 displays the transformed image across several color tolerances, revealing the noticeable difference. The parameters defined to convert the present 2D image into an STL image that is compatible with 3D printers are shown in Table 1. The image's dimensions (155\*155), dominant color for RGB ([15.842275670675301, 12.503171666446413, 26.4560371789789]), and dominant color rate

(0.90359431596545) are determined after reading the image file, the image's width in pixels, and the image's color threshold for color tolerance. The image's shape is then determined, as shown in Figure 6. We find the image threshold (155 is provided), the STL file base Z-input (16), and the STL shape Z-input (31) in the second phase of the trials. As shown in Figure 7, the STL file is then prepared with regard to the Z-dimension.

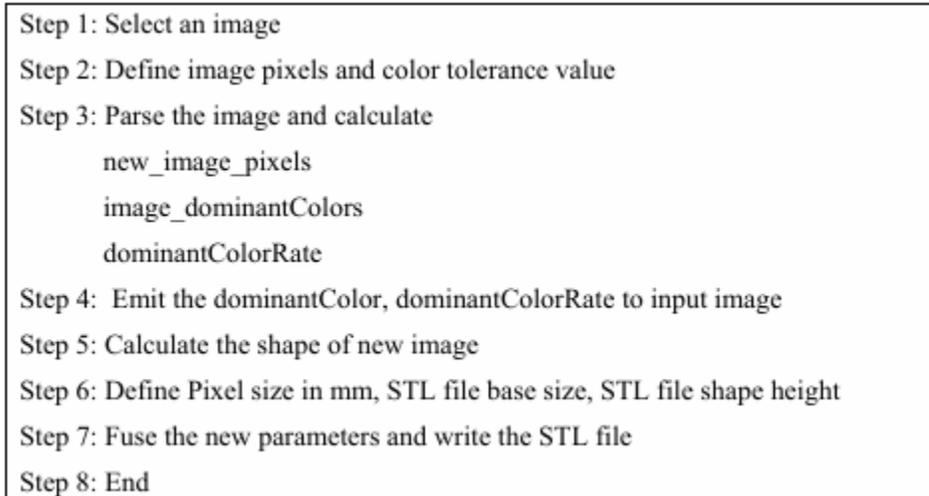


Fig. 4 Algorithm for STL creation



Fig. 5 Original 2D input image [17]



(a) Image with color tolerance 100      (a) Image with color tolerance 70

Fig. 6 Calculated shape

**Table 1** Parameters

| S. no | Name of parameter     | Input value | Calculated value   |
|-------|-----------------------|-------------|--|
| 1     | Image width in pixels | 155         | –  |
| 2     | Image color threshold | 85          | –  |
| 3     | Image dimensions      | –           | 155*155  |
| 4     | dominantColor         | –           | [15.842275670675301,<br>12.503171666446413,<br>26.4560371789789] |
| 5     | dominantColorRate     | –           | 0.90359431596545   |
| 6     | threshold_input       | 155         |  |
| 7     | Unit_Input            | 4.16        |  |
| 8     | Base_Z_Input          | 16          |  |
| 9     | Shape_Z_Input         | 31          |  |

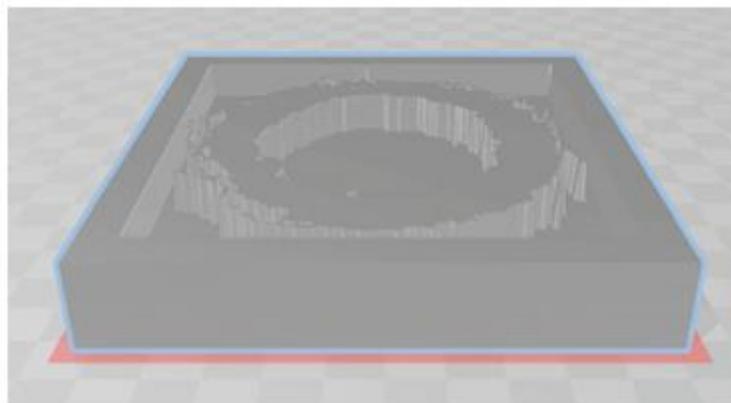


Fig. 7. 3D-STL output from the input image

### 3. CONCLUSION

There are two stages to the process. The first involves reading a 2D image file and defining determine the black-and-white and color tolerance values, as well as the image width of a picture that is being input. As part of the experiment, we shrank the

triangles down to reduce the file size of a picture. We have also determined the dominating color for RGB at the same time and predominate (15.842275670675301, 12.503171666446413, 26.4560371789789)). 0.90359431596545 is the color rate. During the second stage of STL development, we have image's z-base according to the STL

threshold value, input for the image's Z-shape where the corresponding values were 16, 31, and 4.16. The method that has been suggested is carrying out in only 8  $\mu$  seconds, and it performs better than [3-5]. Given that Since STL files do not save image color data, we will need to find a way to include it in our future projects aim to generate an STL file with color information.

#### **REFERENCES:**

1. A review by Mohammad T. et al. (2014) on the topic of build orientation in multilayer manufacturing. *Internat. J. Manuf. Technol.* 27:47–73
2. Yan X et al. (1996) A survey of systems and methods for fast prototyping. *Compensation and Benefits Department* 28: 307-318
3. STL file analysis (Szilvsi-Nagy M. et al., 2003). "Math Comput Model" 38(7-9): 945-960
4. Data creation of layer slices in FDM manufacturing was discussed in a 2016 study by Zhou H-F et al. *Academic Journal of Engineering and Science*, Volume 105, Pages 419-424
5. Rule for Right-Handed Products. This is the URL for the class 1 right-hand rule: <http://www.physics.udel.edu/~watson/phys345/Fall1998/unit-1.html>. Retrieved January 4, 2021.
6. The family of file formats known as STL (STereoLithography). The link to the digital database is <https://www.loc.gov/preservation/formats/fdd/fdd000504.shtml>. Retrieved January 4, 2021.
7. Homepage for file formats. here: <https://docs.fileformat.com/cad/stl/>. Retrieved January 4, 2021. Fabbers'
8. Standard data formats are 8. The URL for this page is [https://www.fabbers.com/tech/STL\\_Format](https://www.fabbers.com/tech/STL_Format). As of January 4, 2021
9. Guha S et al (2015) Mesh simplification via a volume cost measure. *Int J Comp Graph Anim* 5(2):53–64
10. Talton JO A short survey of mesh simplification algorithms. [https://cg.informatik.uni-freiburg.de/intern/seminar/meshSimplification\\_2004\\_Talton.pdf](https://cg.informatik.uni-freiburg.de/intern/seminar/meshSimplification_2004_Talton.pdf)
11. Cignoni P et al (1998) A comparison of mesh simplification algorithms. *J Comp Graph* 22(1):37–54