Int J Adv Manuf Technol (1999) 15:119–126

The International Journal of Advanced Manufacturing Technology

Development of a Stereolithography File Interface with ArtCAM™

S. Boralingiah[1], C. K. Chua[2], T. Ong[1] and S. M. Chou[2]

[1]Delcam Asia Technical Centre, Singapore; and [2]Nanyang Technological University, Singapore

This paper presents the technique used for converting a 3D ArtCAM™ relief file into the stereolithography (STL) format. This 3D file is obtained from a 2D picture using a 3D CNC engraving software called ArtCAM™.

The problem is to convert the 3D relief files into an STL format without compromising the accuracy and details of the relief. This paper discusses the size of the files which have been converted, and steps taken to reduce the file size by reducing the number of triangles or facets in the STL file. The discussion involves the verification of the converted model by comparing it with the original relief. The problems associated with the reduction of triangles when the file has been converted to STL format are also discussed.

Keywords: Artwork; CNC; Relief format; Stereolithography apparatus; STL

1. Introduction

ArtCAM™ is a 3D CNC engraving software produced by Delcam International and is used to convert a 2D picture into a 3D relief format. The 2D picture can be a scanned picture in bitmap format, or any picture file in graphics format such as BMP, TIFF, GIF, JPEG or PCX format. The software converts this picture into a 3D format (file extension *.rlf) by colouring the picture and assigning different colours to each part of the picture. The colours are then given an altitude or height so that when a relief is calculated and displayed, each of the colours is transformed into a relief and the whole image is viewable as a 3D format called the relief. The output relief format (*.rlf) is specific to the ArtCAM™ software which is used for CNC engraving. However, the relief format is not suitable for rapid prototyping systems. In order to create the 3D part using rapid prototyping technology it is necessary to transform the relief format into a STL file.

Rapid prototyping (RP) is a key technology of the 1990s. More than two dozen RP techniques have emerged since the first RP technique, stereolithography, was commercialised in 1988 [1]. The most commonly used input to a rapid prototyping (RP) system is the *de facto* stereolithography file (STL). All vendors of RP systems accept this format and practically all major suppliers of CAD/CAM systems today provide an interface between their CAD model and the STL file.

Before describing the problem in detail, a brief introduction to ArtCAM™ and the relief file is given. The ArtCAM™ process begins with line artwork. This can be scanned into the system, imported from other Windows applications or created directly with the system's Paintbox style functionality. Sophisticated graphics editing tools enable artwork to be cleaned and manipulated quickly and easily.

The artwork is coloured by the use of brushes and flood fill tools, and each colour is then associated with an individual shape profile. The parameters of these profiles may be controlled by defining the basic shape, start height, limit height and wall angle, giving the user almost total control over a wide range of 3D effects. A relief is then calculated for each colour, which has an assigned shape profile. Adding or subtracting reliefs may achieve more complex results. The resulting relief may be viewed, zoomed and rotated about any axis – a useful feature for approving designs with customers. Figure 1 shows an example of a bear as a Windows bitmap. Figure 2 shows the bitmap file converted into a relief image.

2. Format of Relief and STL Files

The formats of the relief and STL file, which are the input and the output files respectively, are discussed in detail. The structure of their internal detail is explained with the help of figures.

2.1 Format of Relief File (Input File)

A relief file consists of a 3D image in *x*-, *y*- and *z*-coordinates. The *x*- and *y*-coordinates represent the in-plane data, and the *z*-coordinate represents the height measurements in pixels. The 3D relief image is bounded by a rectangular frame. The height

Correspondence and offprint requests to: Dr C. K. Chua, School of Mechanical and Production Engineering, Nanyang Technological University, Nanyang Avenue, 639798 Singapore. E-mail: mckchna@ntuvax.ntu.ac.sg

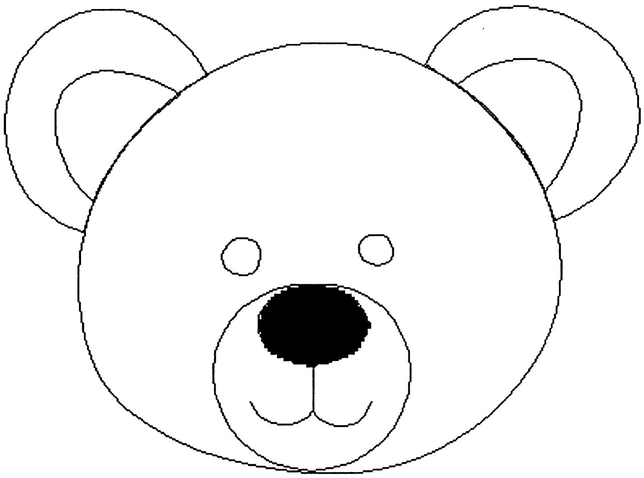

Fig. 1. The bear as a bitmap picture.

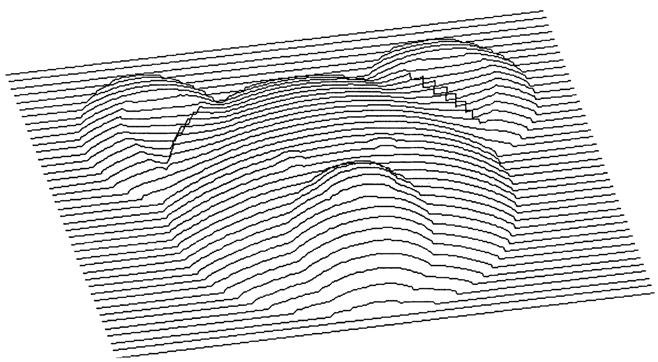

Fig. 2. The bear as a relief image converted from the bitmap picture using ArtCAM™.

of the pixels gives the *z*-coordinates, which is the most important part of the relief. The coordinates represent the internal structure of the relief. The file extension is .rlf. The contents of the relief file can be binary or ASCII.

A relief form is represented internally as a 2D array of 16 bit signed integers along with a scaling factor used to transform the integer value into a floating-point height. This representation halves the memory requirements of a relief form when compared to storing the values as integers.

2.2 Format of STL File (Output File)

The STL file format is the most commonly used file format for input in rapid prototyping technologies and is also the *de facto* industry standard. The STL file defines the surface of an object as a set of interfacing triangles or facets.

Each facet as shown in Fig. 3 is defined with three vertices and a normal, which identifies which side faces out and which side faces in. In the STL file, solid models are represented as an unordered collection of facets and each facet has an outward directed facet normal associated with it [2].

The generation of these facets depends on the information contained in the STL file. It should be noted that in the format of the STL file, the coordinates of the vertices are ordered

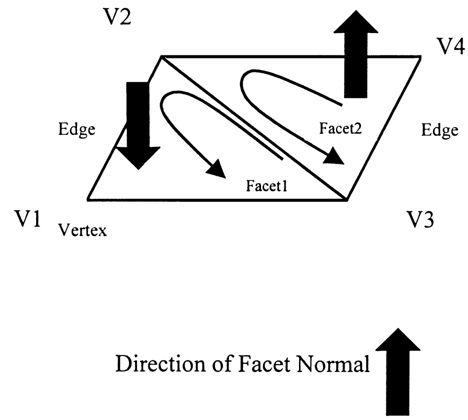

Fig. 3. Description of facet.

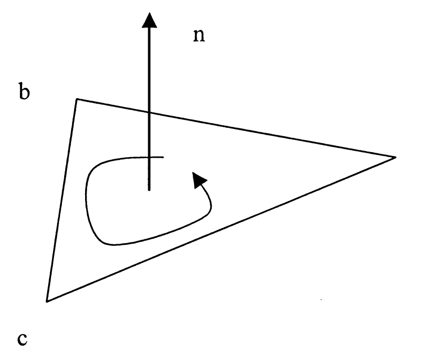

Fig. 4. Righthand screw rule.

according to the righthand screw rule, that is in an anti-clockwise direction such that the normal of the facet is directed away from the model as shown in Fig. 4. Other important information that can be derived from the STL file is that for every facet edge, there must be one other facet sharing the same edge. Since the vertices of a facet are ordered, the direction on one facet's edge is exactly opposite to that of another facet sharing the same edge. This necessary condition is known as the Mobius rule, as shown in Fig. 5. A facet can reference the three edges, which bound it [3]. Each edge can

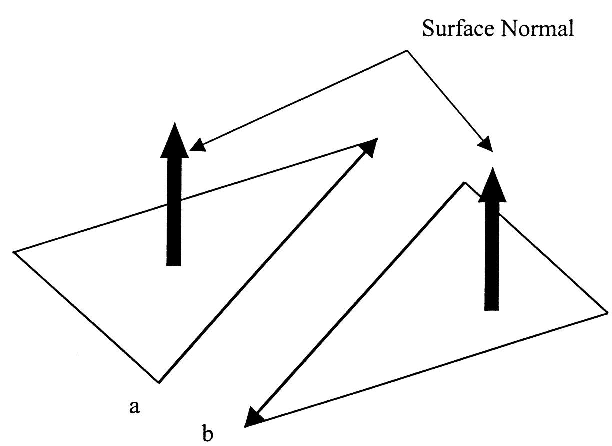

Fig. 5. Mobius rule – edge shared by two facets.

reference the two vertices, which define it. Vertex points can contain the connectivity information to all edges or faces, which share it. The STL format contains facets only with the minimum information necessary to define the image or solid object.

For each vertex that is present in the STL file it is necessary to calculate the normal, in order to determine which way the facet is facing, whether inwards or outwards. To calculate the normal, the essential information required is the vertices which bound the facet. The normal is calculated by the cross-product of the vertices.

In order to understand the format of an STL file it is important to know the basic internal structure of an STL file. There are two different formats for STL files. One is the ASCII format, which is human readable, and the other is binary, which is totally unreadable. During the developmental stages of this project, the ASCII format was used for debugging purposes.

The binary format is used in the final release version because of its compactness. To illustrate with the example of the bear, the STL file in ASCII is 12 megabytes whereas the STL file is only 900 kilobytes in binary. The size of the file differentiates these two formats.

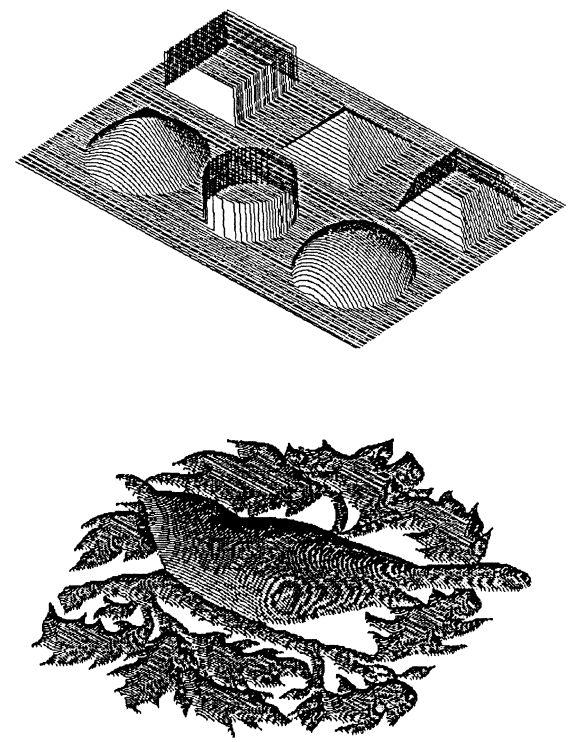

Fig. 6. Low and high resolution relief.

3. STL Conversion

To convert the relief file into STL output, a number of problems must be overcome. They are mainly related to the size of the image file being converted. The major concerns that will affect the image size are the resolution of the relief image (input file), the size of the output file with respect to testing, and the reduction of triangles which are directly dependent on the resolution of the image. Each problem will be explained in detail as follows.

3.1 Limitation of the STL Format

In order for the file size not to be too large, the ASCII version of the STL file was used only for verification purposes but the binary version was used for testing and in the final release version. The size of the converted STL file (binary) should not exceed 50 MB. This is one limitation of the STL format and there are several other disadvantages. This is one of the major problems affecting the testing of the file, because if the file cannot be tested, it is not possible to detect errors. This puts a major constraint on the image size of the relief file. The example used for testing purposes was that of the face of a bear. The output of the STL ASCII file size was 12 megabytes whereas the size of the binary file was 900 kilobytes.

3.2 Resolution of the Relief Image

The resolution of the relief image also affected the output. If the resolution of the relief image was greater, then the STL output would grow in direct proportion to the resolution of the relief image. The primary reason being that, for a high-resolution image the number of pixels used to describe the image would be far greater than required. Hence in order to keep the size of the file under control, it is necessary to keep the resolution of the bitmap image, used as input to the ArtCAM software, to an acceptable level of visibility. An acceptable level of visibility means that the bitmap image can be viewed satisfactorily without increasing the resolution measured in dots per inch (DPI). The number of DPI has to be as small as possible, but at the same time the image detail should be clear. Once this has been done the relief obtained from the bitmap image will also be of reasonable resolution. This will help in reducing the size of the output STL file. An example of a relief with low resolution and high resolution is shown in Fig. 6.

3.3 Triangle Reduction

The next factor affecting the size of the output file is the number of triangles used in describing the STL output. The technique used in reducing the number of triangles was by searching for triangles having the same pixel height. Then, these triangles with the same pixel height are grouped in an orderly manner and reduced to only two triangles. An algorithm in the next section explains the implementation of this step. Thus, reducing the STL output file to fewer triangles, results in fewer points in the STL output file, thereby reducing its overall size.

4. Conversion Algorithm

The algorithm has the following steps:

1. Converting each set of points on the relief into triangles. Checking for points with the same pixel height. Formation of triangles with the same pixel heights.
2. Checking for gaps in the relief image [4].
3. Formation of box so that the STL output is closed with a variable height.

The relief is first treated as a whole with rectangular mapping of points as shown in Fig. 7 as for an *XY* graph. The points on the *XY* graph represent the pixels (*Z* plane). Each pixel may or may not have a height, depending on its location on the image. The direction of its traversal path starts from the origin zero. The following steps are carried out using the notation (*x*, *y*):

1. Start with the first point (0, 0) and its counterpart (0, 1) which is on the top left. These two points are considered as a set of points for comparison.
2. Select the next point along the *x*-axis (1, 0) and its counterpart (1, 1) which is on the top right. These two points are considered as the next set of points.
3. Check if the heights of the first and second sets of points along with their counterparts are equal.

There are three possible cases that arise from the comparison.

1. If the heights of the points are equal, then there are two possibilities to be considered. The first possibility is the heights of the pixels for these points is zero. Triangles are not formed for this possibility because this part of the image does not contain the relief points. The second possibility is that the heights are equal but the heights of the pixels are greater than zero. If the heights are equal and greater than zero, a comparison with the next point in sequence is made. Here the next point is the third point. A comparison is made between the first and the third set of points. A continuous comparison is made until the end of the *x*-axis is reached or when the heights are not equal.
2. If the heights are not equal then triangles are formed as shown in Fig. 8.
3. Once a set of triangles is created between two sets of points *A* {(0,0),(0,1)} and B {(1,0),(1,1)} as shown in Fig. 8, then the second set of points in this case *B* {(1,0),(1,1)} is used as the reference origin from which the heights are compared with the next set of points.
4. This is done for the entire image, covering the entire *x*- and *y*-axes.

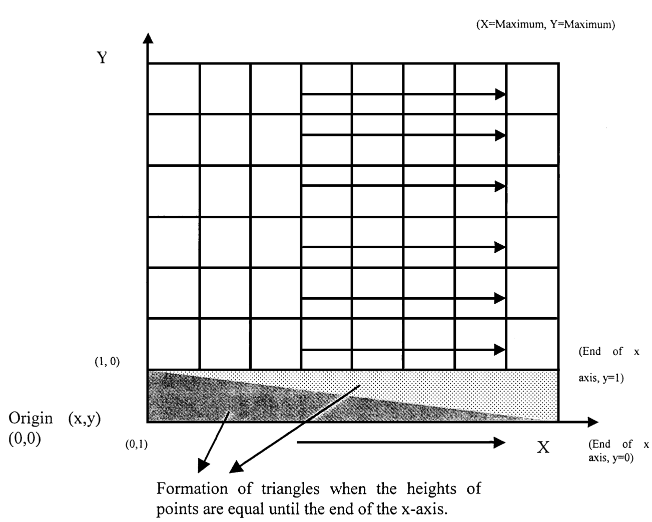

Fig. 7. Case A, traversal path showing formation of triangles when heights of two consecutive points are equal.

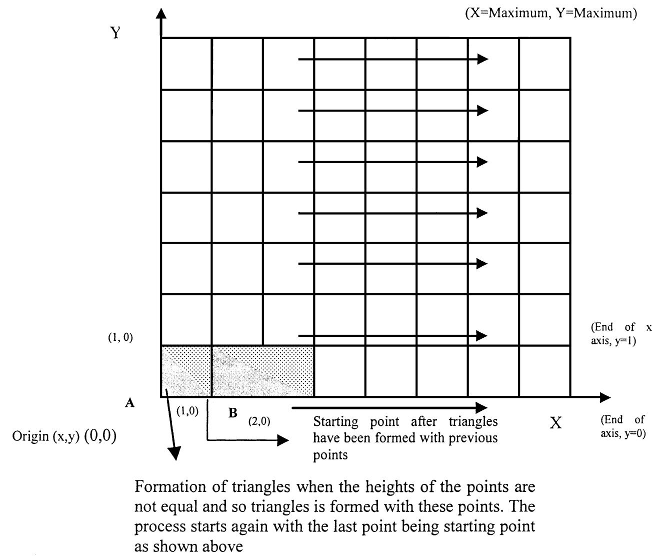

Fig. 8. Case B, traversal path showing formation of triangles when heights of two consecutive points are not equal.

The next step involves detection of gaps for those areas of images where the surface of the image protrudes outwards with its upper surface not connected to the base of the relief on all sides. One side of the surface is open whereas the other side is connected to the relief as shown in Fig. 9. The above algorithm does not handle this kind of problem and so the entire relief image is scanned for such gaps and if they exist, these gaps are covered by the formation of triangles using the triangle reduction technique.

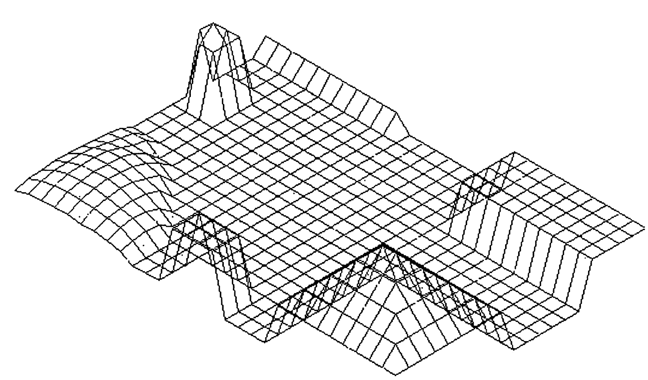

Fig. 9. Gaps in relief image.

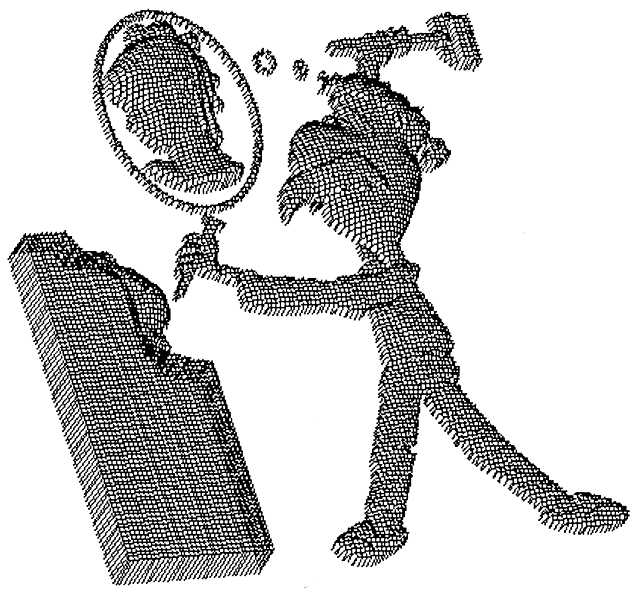

Fig. 10. Image of the sculptor shown in STL format without rectangular box.

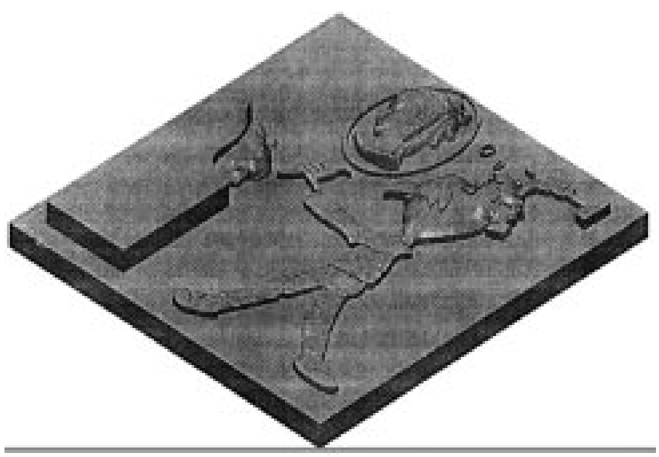

Fig. 11. Shaded image of the sculptor converted to an STL file enclosed by a rectangular box.

The final step is the formation of a box. Giving a fixed height measurement to the final image does this. The image is enclosed by a rectangular boundary based on the relief image in pixels. An exact shaped boundary is provided at the bottom with enclosures on all four sides. The final image is a box with fixed width with the STL image located on the top surface of the box. Figures 10 and 11 show the sculptor model before and after the formation of the box, respectively. This is done so that the whole relief image is closed and no openings are shown.

5. Verification

The verification of the converted STL file is carried out by a visual check and a physical check through the SLA model.

5.1 Visual Check

In this method, the converted STL file can be viewed using any CAD/CAM software that is capable of viewing STL files. A close examination should reveal any gaps or unclosed triangles.

5.2 Physical Check

In this method, the actual binary STL file is fed into a stereolithography apparatus (SLA). The converted file is fed into a computer where the model is sliced into cross sections or layers. Liquid resin is used as the initial material to form a solid version of the image, which has been converted from the relief form into STL format. Before carrying out the physical check it is advisable to go through an extensive check of the STL file using the visual method.

6. Building of Model by the SLA

The SLA process chamber consists of a vat containing liquid polymer photopolymer resin, a platform on which the object is to be built and whose height is controlled by an elevator mechanism, a re-coating blade wiper and a helium–cadmium or argon laser subsystem. At the start of the object building process, the platform is positioned at a depth of one layer thickness below the resin level. The laser traces over the areas of the resin surface defined by the vectors at the cross-section of the first layer. The area where the resin is struck by the laser beam solidifies to form the first layer of the object. Subsequently, the platform is lowered by a distance equal to the layer thickness and the wiper sweeps across the surface, to prepare the construction of the next layer as the process repeats itself. When the object has been completely built, the platform is raised above the vat of resin to drain off the excess liquid resin that has adhered to the object.

6.1 Different Methods in Building an SLA Part

There are three methods by which an SLA part can be built. The general process of building an SLA part has been explained earlier. Each method is explained in detail with respect to the resolution of image. The advantages and disadvantages of each method are also highlighted.

Before explaining each of these methods in detail, the resolution of the 3D relief with respect to ArtCAM™ is explained. The relief image is composed of pixels having a *z* height in the *x*-, *y*-plane. The difference in height between each layer of pixels is approximately 0.02 mm. This information is obtained from the internal detail of the relief. The resolution of any model built with the SLA varies from 0.10 mm to 0.125 mm in the *z*-plane. This is due to the fact that the motor drives the *z*-axis of the SLA. The SLA can use either of two types of motor for building the parts. The first type uses a servomotor whereby the distance between any two layers can be varied, and hence the model can be built with a better resolution. Servomotors are expensive and so the second type

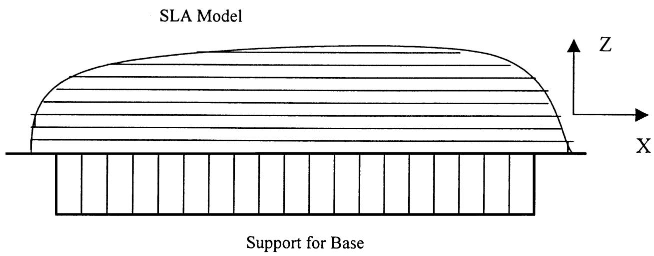

Fig. 12. SLA part built using the horizontal method.

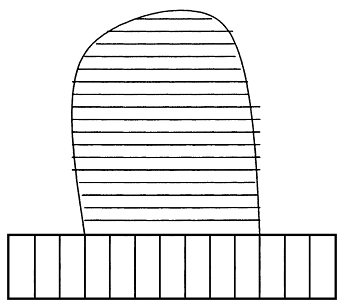

Fig. 13. SLA part built using the horizontal method.

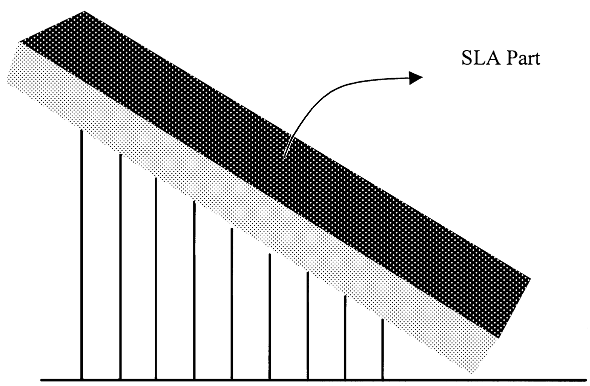

Fig. 14. SLA part built using the slanting method.

Fig. 15. SLA part of the sculptor.

Fig. 16. SLA part of the eagle crest.

with a stepper motor is more widely used. In the case of the stepper motor, the distance between any two layers is fixed, and varies from 0.10 to 0.125 mm. The limitation of using this type of motor is that if a detail exists on the 3D relief with a tolerance of 0.02 mm, and the tolerance of the SLA using the stepper motor type is between 0.1 and 0.125 mm, it can be seen that the detail resolution from the SLA part is five-times worse than on the original 3D model. This technique is common, irrespective of the orientation of the model, be it vertical, horizontal or slanting. All SLA parts are built from the base or bottom up.

6.2 Building SLA Part Using the Horizontal Method

In this method, the part to be built is located with its base as shown in Fig. 12. The supports are located below the model. The base of the model is constructed first and then subsequently the next level is built on the most recent level. The distance between each layer is equal to the layer thickness, which is fixed as the step height specified in the SLA. In this method, complete detail of the model cannot be obtained because of the five-fold difference in the tolerance level between the SLA machine and the 3D relief. Much of the detail is lost. This method can be used only when the relief image has little detail.

6.3 Building an SLA Part Using the Vertical Method

In this method, the model is constructed vertically. The position of the model is as shown in Fig. 13. The model is built with one of the sides being fixed as the base. The layers are formed starting from the side of the relief image. This method can be used only when the model being built is big and thick. The

Fig. 17. ArtCAM™ image of Merlion.

Fig. 18. Relief image of Merlion.

Fig. 19. SLA part of the Merlion.

disadvantage of using this method is that for small and thin models when the upper part is constructed the liquid resin that flows to form each layer may topple the model as it is being built. The reason is that less force is required to topple the upper part of the component than the lower part.

6.4 Building SLA Part Using Slanting Method

This method is used in order to solve the problems faced by the vertical and the horizontal methods. Owing to the slant orientation, supports are required as shown in the Fig. 14. Depending on the size of the model, the support can be varied in design (web, cylindrical or other type) and quantity (number of support elements) to provide adequate support to the model. This method is far superior to the other methods for providing a better tolerance than for the first method. It also strikes a good compromise on building time between the first two methods. Figures 15 and 16 show SLA parts of the sculptor and the eagle crest, respectively, built using this method.

6.5 Case Study: Making of SLA Part – Merlion Model

In this case study, the process of making a 3D model of a merlion using ArtCAM™ is illustrated. The first step involves scanning the picture and saving it into a Windows graphic file format. After the picture has been scanned, the picture is displayed using ArtCAM™ and the picture is assigned different colours. Each of these colours is assigned a different height. Once this has been done, the 2D image is transformed into a 3D-relief image. This image is then converted into the STL format. The image obtained in STL format contains only the side view of the merlion. The STL file is opened using CopyCAD, a CAD software used in reverse engineering. CopyCAD is a powerful reverse engineering system that allows the creation of 3D CAD models from existing parts or physical models. The software provides comprehensive tools for the creation of CAD surfaces from digitised data. Digitised data from conventional coordinate measuring machines, tracing machines and laser scanners can be imported, edited and accurately surfaced and inspected. Using CopyCAD, the relief that contains the base that is not part of the image is removed leaving only the main image. The final image obtained is a side view of the merlion. In order to construct the full image, a mirror image of one side is patched on to the other side to form a full 3D image of the merlion. The 3D image is then converted into an SLA part using the vertical method. The vertical method is used because the model is thick and large. Figures 17, 18 and 19 show the various steps by which the 3D merlion was built from the 2D scanned image.

7. Conclusion

The conversion of a relief file into STL format is specific to ArtCAM™ software. The size of the converted file is directly dependent on the image size. Since the image size is dependent on the resolution of the image measured in dots per inch, as the image size increases the size of the converted STL file increases. If the image has more detail, the number of triangles is very large. Reduction of the number of triangles, where possible, is thus critical. Reduction of the number of triangles can be exploited if the image has points having the same height on the *z*-plane. The above technique has been implemented and delivered in the new version of ArtCAM™ released in July 1997.

Acknowledgements

We wish to acknowledge the sponsorship of this research and development project by Singapore's National Science and Technology Board. This project is a joint development in collaboration with Delcam International Plc, Gintic Institute of Manufacturing Technology and Nanyang Technological University's School of Mechanical and Production Engineering.

References

1. C. K. Chua and K. F. Leong, "Rapid Prototyping: Principles and Applications in Manufacturing", John Wiley, 1997.
2. C. T. Lee, "Implementation of repair algorithms for faulty stereolithography files", Final Year Project School of Mechanical and Production Engineering, Nanyang Technological University, Singapore, 1996.
3. Stephen J. Rock and Michael J. Wozny, "Utilizing topological information to increase scan vector generation efficiency", Solid Freeform Fabrication Symposium Proceedings, the University of Texas at Austin, Texas, pp. 28–36, 12–14 August 1991.
4. K. F. Leong, C. K. Chua and Y. M. Ng, "A study of stereolithography file errors and repair. Part 1. Generic solution", International Journal of Advanced Manufacturing Technology, 12, pp. 407–414, 1996.