Heterogeneous Object Modeling for Rapid Prototyping

Xiaojun Wu
Harbin Institute of Technology Shenzhen Graduate School
P.R.China

1. Introduction

A heterogeneous object is referred to a solid component consisting of two or more material primitives distributed continuously or discontinuously within an object. Modeling and manufacturing of heterogeneous object (HO) have been paid much attention recently as the advent of rapid prototyping manufacturing technology, which makes it possible to fabricate the heterogeneous object. As the continuously variation of material composition produces gradient in material properties, they are often known as functionally gradient materials (FGM), shown in Fig. 1(a). For example, a component contains two compositions, metal and heat resistance material (such as ceramic); the material distribution is illustrated in (b). From the figure we can see that metal increases its fraction gradually from one side to another (the red line), while the heat resistance material linearly reduces its fraction (the green line), which can avoid the stress concentration because of the thermal stress relaxation in transition of two materials, shown in (c). A discontinuous change in material composition generates distinct regions of material in the solid, which is usually called multi-material object (MMO) such as composite materials, as demonstrated in Fig. 2 (ZCorp (2005)). MMO has been extensively used in industry for a long time, while FGM has shown tremendous potential in many fields, such as aeronautics and astronautics, biomedical engineering, and nano-technology, etc.
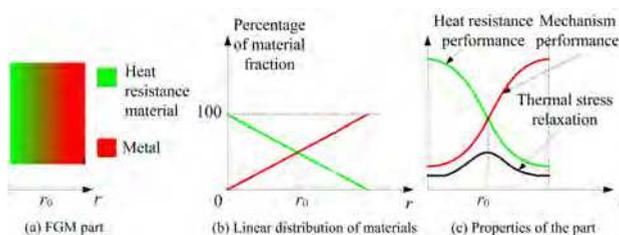

Fig. 1. Model of functionally gradient material.

Rapid prototyping can offer capability to fabricate the component with material variations because of the characteristic of layer by layer manufacturing. Shape deposition manufacturing (SDM) of Carnegie Mellon University and Stanford University and laser engineered net shaping (LENS) of Sandia National Laboratory (SNL) has produced FGM parts with variety of metallic powders (Binnard (1999),Fessler (1997), Jakubenas (1998), Ensz (2002)). With 3D printing, Massachusetts Institute of Technology has created fine ceramic FGM components.

Fig. 2. A multimaterials blade.

Just as the color jet/laser printer can produce colorful pictures by using halftone technique in 2D, the heterogeneous object can also be fabricated by a 3D printer which is regarded as the most suitable means to create HO. Readers can also refer to the products of Z-Corp (ZCorp (2005)).

In order to take full advantage of the greatest potential of heterogeneous objects, one must have matching capabilities for their computer modeling, analysis and design optimization. The primary focus of the recent research and development in these fields are on the computer representation schemes for heterogeneous objects, by extending the mathematical models and computer data structures of the modern solid modeling techniques to include discrete material regions of interfacial boundaries and heterogeneous properties.

To build an object by a particular rapid prototyping technology, certain paths must be generated to guide the executor to add the material layer by, called path planning (Marsan (1997)). It is necessary to slice the model into contours through a serial of parallel plane to intersect with the object. Conventionally, we do not consider material information when doing slicing operation. But the heterogeneous object can not be dealt with the same strategy. Siu et al. proposed a contour sub-division algorithm on each layer from slicing a heterogeneous object where the material gradient is decomposed into sub-contours according to the different grading variation (Siu (2002a), Siu (2002b)). However, the "grading step-width" based method separates the continuous material domain into discrete strips on each slice.

Approaches for modeling and representing a heterogeneous object have been extensively studied in the computer and manufacturing community in the last decade. Kumar and Dutta proposed an approach to model multi-material objects based on R_m sets and R_m classes primarily for application in layered manufacturing. Boolean operators to facilitate the modeling procedure were defined (Kumar (1997), Kuman (1998)). Jackson et al. proposed a local composition control (LCC) approach to represent heterogeneous object in which a mesh model was divided into tetrahedrons and different material compositions were evaluated on the nodes of the tetrahedrons by using Bernstein polynomials (Jackon (2000), Liu (2000)). Chiu et. al. developed material tree structure to store different compositions of an object (Chiu (2000)). The material tree was then added to a data file to construct a modified format suitable for RP manufacturing. Marsan and Dutta presented a method to model material properties in the form of tensor product surfaces within the framework of heterogeneous solid modeling (Marsan (1998)). Siu and Tan developed a scheme named 'source-based' method to distribute material primitives, in which any material within an object could have varying properties (Siu (2000)). The feature-based modeling scheme was extended to heterogeneous object representation. By controlling boundary conditions of a virtual diffusion problem in the solid, designers could control its material distribution (Qian (2003), Qian (2004)). Kou and Tan suggested a hierarchical representation for heterogeneous object modeling by using B-rep to represent geometry and a heterogeneous feature tree to

express the material distributions (Kou (2005)). Samanta suggested a scheme to represent and design heterogeneous object by using a free-from functions to describe complex shapes of geometry and material features (Samanta (2004)). Zhang et al. addressed the problem of heterogeneous material object modeling and analysis through a constructive approach (Zhang (2004)). Wu et al, developed a heterogeneous object modeling system based on volumetric date set (Wu (2007)). Various methods for designing and optimizing objects composed of multiple regions with continuously varying material properties have been developed. Wang and Wang proposed a level-set based variational scheme (Wang (2004)). Biswas et al. presented a mesh-free approach based on the generalized Taylor series expansion of a distance field to model a heterogeneous object to satisfy the prescribed material conditions on a finite collection of material features and global constraints (Biswas (2004), Tsukanov (2003)). Kou et.al proposed a heterogeneous object modeling system which employed the B-rep to represent the geometry shape and the heterogeneous feature tree (HFT) to represent the gradient material information, and then a virtual manufacturing prototype system was created based on voxelization of the heterogeneous object (Kou (2006)), shown in figure 3. Hu, et.al. addressed the design problem of processing time of manufacturing 3D heterogeneous objects (Hu (2008)). In their method, the processing time was considered as a optimization variable to design and manufacture 3D heterogenous objects through using self-adaptive and real-valued evolutionary strategy.
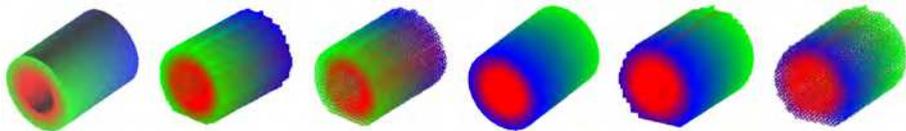

Fig. 3. HFT based heterogenous object and voxelized model (Kou (2006)).

However, almost all of the research interests are mainly focused on the computer representation of heterogeneous object, rather than the whole procedure for rapid prototyping fabrication of heterogeneous object. Almost all of the previous proposed approaches were verified in commercial software packages (Siu (2000), Liu (2004)), such as Solidworks, Unigraphics, etc. In Kou (2004), a commercial CAD package independent system is developed to deal with the HO modeling, but not including the slicing procedure for RP manufacturing. Kou. et.al. in Kou (2007) given a good survey for the different heterogenous object modeling system. Readers can refer to (Kou (2007)) for a whole review of modeling heterogenous object. In this chapter, we will introduce a heterogeneous object modeling system based on volumetric dataset to address a complete design pipeline from CAD model to heterogenous object for rapid prototyping.

In realistic world, the interior of every object is defined homogeneously or heterogeneously, instead of a shell with zero thickness. With the fast development in computer hardware, especially faster, larger and cheaper memories available, computer graphics are being transformed from surface based to volume based, just like the transition from vector graphics to raster graphics in the seventies (Kaufman (1993)). One of the most outstanding features of volumetric dataset is its capability to represent the inner structures of an object such that measurable properties, such as material, color, density, and strength, can be associated to each voxel. Therefore, it is a perfect choice to utilize volumetric dataset to describe the internal properties or structures of a heterogeneous object. In fact, voxel-based models are exploited for part modeling, analysis and manufacturing (Chandru (1995)). In our HO

modeling scheme we take volumetric dataset as a carrier of material primitives, while the shape of the object is described by the geometric model. So it is convenient to manipulate the dataset and implement the boolean operations (e.g. union, difference, intersection, etc.), and the volumetric model can be easily processed to generate 2D slices which are essential to manufacturing with rapid prototyping techniques. In our approach, we develop a new method to slice a heterogeneous object where the geometric contours are taken as constraints to resample the heterogeneous information through pixel overlap interpolation and trilinear interpolation strategy. This method can maintain the original heterogeneous material information as much as possible and improve the slicing boundary as accurate as possible. This method is called material resampling with geometric constraint (MRGC). The output of our scheme is bitmap liked raster image with sharp boundary which is very suitable for the 3D printer based heterogeneous object fabrication devices.

2. Volumetric dataset based heterogeneous object modeling

2.1 Mesh model voxelization

Voxelization procedure converts a geometric model into volumetric dataset. As a matter of fact, volumetric dataset comes from a variety of fields, such as human organs scanned by Computer Tomography (CT) or Magnetic Resonance Imaging (MRI), the visual human project, scientific computation or simulation, computational fluid dynamics (CFD), meteorology, seismic exploration, etc. These datasets can be organized into Cartesian, regular, rectilinear, structured, unstructured and hybrid data format. In the past decades, a lot of methods on voxelization have been developed (Kaufman (1987), Jones (2000), Kong (1989)). Most of the voxelization methods are an extension of the classical scanning conversion algorithm from 2D to 3D. In our HO representation, we develop a voxelization algorithm to convert geometric models into volume dataset. For simplicity, we only utilize polygonal meshes (triangular meshes) to describe the voxelization algorithm, but all geometric models can be voxelized, such as CSG model, freeform surfaces, implicit or explicit surfaces (Huang (1998), Adams (2004)). The algorithm is described briefly as follows.

Let S be a plane in 3D space, G and H be two planes parallel to S and locate opposite sides of S shown in Fig. 4. Their functions are expressed as Equations 1 and 2.

$$Ax + By + Cz + D = 0 \tag{1}$$
$$Ax + By + Cz + D + t = 0 \tag{2}$$

If plane S would be voxelized, just let the distances from the points between plane G and H satisfy Inequality 3

$$-t \leq Ax + By + Cz + D \leq t \tag{3}$$

where t is defined as $t = t_6 = (L/2)\cos\beta$ if s is 6-adjacent voxel plane, and $t = t_{26} = K\cos\alpha = (L/2)\cos\alpha$ if S is 26-adjacent voxel plane. The definition of α, β, L, K and N are shown in Fig.5. We use t_6 and t_{26} to replace t in inequality 3, then two theorems can be induced.

Theorem1 and Theorem2 are suitable for the voxelization of an indefinite plane. In practice, the primitives, such as vertices, edges and faces, should be processed respectively for speedup the calculation. The sets of $\tilde{S}_v$, $\tilde{S}_e$ and $\tilde{S}_b$ represent the voxel sets of vertexes, edges and facets respectively. An object's voxel representation can be obtained from $\tilde{S} = \tilde{S}_v + \tilde{S}_e + \tilde{S}_b$. Taking a triangular facet as an example, say S, for each vertex of S we construct a sphere whose center is the vertex and the radius is R_c defined as $R_c = L/2$ when 6-adjacent and $R_c = (\sqrt{3}/2)L$ when 26-adjacent, showed in Fig.6(a). All the voxels within the sphere belong to set $\tilde{S}_v$. Similarly,
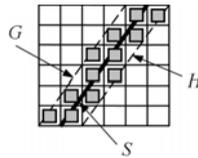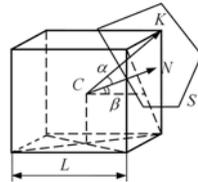
Fig. 4. Fig. 6 Rasterizing Line *S*.

Fig. 5. The definition of α and β.

for each edge of *S*, a bounding cylinder of radius R_c and length *L* is defined, where L is the length of the corresponding edge, seeing Fig. 6(b). All the voxels inside the cylinder belong to set $\tilde{S}_e$. Thirdly, a bounding triangular box opposite to *S* is constructed with two *S*'s parallel planes *G* and *H* and three planes, say $E_i(i = 1, 2, 3)$, perpendicular to *S*, showed in Fig. 6(c). The voxels belonging to the box represent the voxelization of the triangle *S*.

Theorem 1. *Plane S is defined by A, B, C and D, the set $\tilde{S} = \{(x, y, z) | -t_6 \leq Ax + By + Cz + D \leq t_6\}$ defines a 6-adjacent voxel representation of S.*

Theorem 2. *Plane S is defined by A, B, C and D, the set $\tilde{S} = \{(x, y, z) | -t_6 \leq Ax + By + Cz + D \leq t_{26}\}$ defines a 26-adjacent voxel representation of S.*
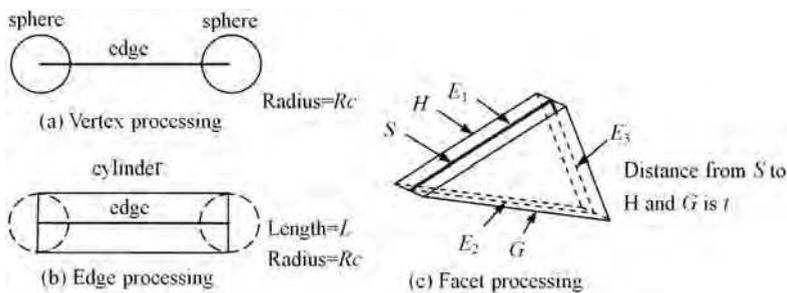

Fig. 6. The voxelization of vertex, edge and facet.

An object is voxelized into volumetric dataset with different resolutions illustrated in Fig. 7 where (a) is a surface model, (b), (c) and (d) are the corresponding volumetric datasets with resolution (64,64,45), (128,128,89) and (256,256,176) respectively. From (b), (c) and (d) we can see that the higher the resolution of the volumetric dataset, the more accurate of the voxel based model to approximate the surface model.

2.2 Evaluation of heterogeneous material

As described above, the core issue of heterogeneous object representation is designing a scheme to evaluate gradient or multi-material within a CAD model according to the

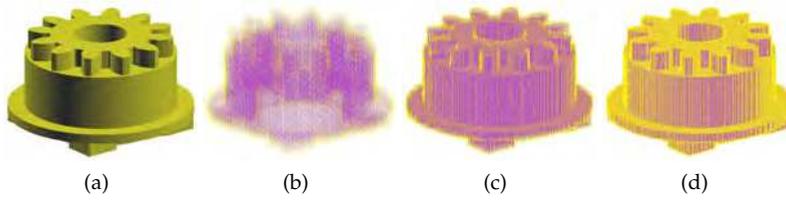| (a) | (b) | (c) | (d) |

Fig. 7. A surface model and its volumetric dataset in different resolutions, (a)Mesh model (b) Resolution (64,64,45) (c) Resolution (128,128,89) (d) Resolution (256,236,176)

specification of users. In our scheme, we exploit the geometric model to describe the shape information. In terms of material information, we use the framework proposed by Kumar (Kumar (1997), Kuman (1998)) to describe material composition in terms of material space which is a vector space and whose components are material primitives, for example, V^3 is a three dimensional material space constituted by three material primitives. The material space is notated as V^m with m material primitives.

Suppose Ω is a subspace of E^3 and $G_i(i = 1, 2, \cdots, k)$ are subsets of Ω. G_i satisfies the requirements 4.

$$\begin{cases} (G_1, G_2, \cdots, G_k) \in \Omega \\ G_1 \cap G_2 \cap \cdots \cap G_k = \Phi \\ G_1 \cup G_2 \cup \cdots \cup G_k = \Omega \end{cases} \tag{4}$$

Defining space B is a subspace of V^m which is a material space with dimensions of m. Let $B_i(i = 1, 2, \cdots, k)$ are subspace of B, which is defined as a mapping of G_i in V^m. B_i should meet the requirements 5.

$$\begin{cases} (B_1, B_2, \cdots, B_k) \in B \\ B_1 \cap B_2 \cap \cdots \cap B_k = \Phi \\ B_1 \cup B_2 \cup \cdots \cup B_k = B \end{cases} \tag{5}$$

A point in a heterogeneous object can be described as

$$P_v = \{(P_e, P_m)|P_e \in \Omega, P_m \in B\} \tag{6}$$

Then we can represent a heterogeneous object as follow

$$P_{MMO} = \{(\boldsymbol{P}_{ei}, \boldsymbol{P}_{mi})|\boldsymbol{P}_{ei} \in G_i, \boldsymbol{P}_{mi} \in B_i, \boldsymbol{P}_{mi} = \boldsymbol{C}; i = 1, 2, \cdots, N; j = 1, 2, \cdots, m\} \tag{7}$$

$$P_{FGM} = \{(\boldsymbol{P}_{ei}, \boldsymbol{P}_{mi})|\boldsymbol{P}_{ei} \in G_i, \boldsymbol{P}_{mi} \in B_i, \boldsymbol{P}_{mi} = \bigtriangledown f|_{P_{mi}}; i = 1, 2, \cdots, N\} \tag{8}$$

where $\boldsymbol{C}$ is a constant material vector in a single B_i, and $\bigtriangledown f|_{P_{mi}}$ is the gradient determined by material distribution function, and N is the number of sampled points inside an object, namely object voxels. To unify the MMO and FGM into a framework, we divide the object into several areas according to the distance field, notated as G_i whose mapping in material space is B_i. Currently, for a single material feature, only three subdivisions can be defined, denoted as G_{-eff}, G_{eff}, G_{+eff}, and the mapping to the material space is B_{-eff}, B_{eff}, B_{+eff}. These three subdivisions are defined as

1. Negative Constant Material Range(NCMR) G_{-eff}:

$$G_{-eff} = \{\boldsymbol{P}_{ei}|\boldsymbol{P}_{ei} \in \Omega \& 0 \leq d \leq d_{eff}\} \tag{9}$$

2. Material Gradient Range (MGR) G_{eff}

$$G_{eff} = \{\boldsymbol{P}_{ei}|\boldsymbol{P}_{ei} \in \Omega \& d_{-eff} \le d \le d_{eff} + d_{-eff}\} \qquad (10)$$

3. Positive Constant Material Range (PCMR) G_{+eff}

$$G_{+eff} = \{\boldsymbol{P}_{ei}|\boldsymbol{P}_{ei} \in \Omega \& d > d_{-eff} + d_{eff}\} \qquad (11)$$

where d is distance(s) from selected feature(s).

Generally, a material distribution function is needed to determine the material variation within an object, which comes from material designer or expert system of material design. This function takes the distance from inner object point to the selected feature(s) as arguments, and it must be satisfy the requirements of $0 \le f(x) \le 1$ in the material gradient range due to the summation of all the material primitives equal to 1. At the moment, $f(x)$ is single variable function to control the composition variation; any analytical, segmental, linear or nonlinear functions can be taken as material distribution function.

2.2.1 Distance function based material evaluation paradigm

Let vector array $\boldsymbol{M}$ store the variations of materials, and each component of $\boldsymbol{M}$, say $\bar{m}_j$, is a vector in size of m, the dimension of material space. $\boldsymbol{P}_{mi}$ is a point in material space, and $\boldsymbol{P}_{mi}$ is defined by $\boldsymbol{P}_{mi} = \{\bar{m}_j|j = 1, 2, \cdots, m\}$, and $\bar{m}_j$ should meet the following requirement.

$$\sum_{j=1}^{m} \bar{m}_j = 1 \qquad (12)$$

At present, only three material primitives can be evaluated by distance field based approach. The value of $\boldsymbol{P}_{mi}$ in MGR is defined by the following formula

$$\boldsymbol{P}_{mi} = \begin{bmatrix} f(d - d_{-eff}) \\ 1 - (f(d - d_{-eff}) + C) \\ C \end{bmatrix}, i = 1, 2, \cdots, N \qquad (13)$$

where C is an invariable representing a constant material composition in the whole object. In this case, the composition function should be confined to $0 \le f(d) \le (1 - C)$. That the independent variable of $f(x)$ is $(d - d_{-eff})$ rather than d is because distance d is computed from reference feature(s), the material function $f(x)$ meets the condition of $0 \le f(x) \le 1$ in interval $[0, d_0]$.

Defining $\boldsymbol{M}_s$ and $\boldsymbol{M}_e$ are the material vectors in the beginning and the end of material gradient range, which can be offered by designers, but it can not guarantee the continuity from NCMR to MGR and from MGR to PCMR. We can use the following equation to compute $\boldsymbol{M}_s$ and $\boldsymbol{M}_e$ to ensure the continuity.

$$\boldsymbol{M}_s = \begin{bmatrix} f(d_{-eff}) \\ 1 - (f(d_{-eff}) + C) \\ C \end{bmatrix} \text{ and } \boldsymbol{M}_s = \begin{bmatrix} f(d_{-eff}) \\ 1 - (f(d_{-eff} + d_{eff}) + C) \\ C \end{bmatrix} \qquad (14)$$

As the above analysis, the material distribution in B_{-eff} can be defined as

$$G_{B_{-eff}} = \{\boldsymbol{P}_{mi}|\boldsymbol{P}_{mi} \in B_{-eff} \& \boldsymbol{P}_{mi} \in \boldsymbol{M}_s\} \qquad (15)$$

The material distribution in B_{eff} is defined as

$$G_{B_{eff}} = \{\boldsymbol{P}_{mi}|\boldsymbol{P}_{mi} \in B_{eff}\} \tag{16}$$

The material distribution in B_{+eff} is defined as

$$G_{B_{+eff}} = \{\boldsymbol{P}_{mi}|\boldsymbol{P}_{mi} \in B_{+eff}\&\boldsymbol{P}_{mi} \in \boldsymbol{M}_e\} \tag{17}$$

Thus, a HO model can be defined as

$$G = ((G_{-eff}, G_{B_{-eff}}), (G_{eff}, G_{B_{eff}}), (G_{+eff}, G_{B_{+eff}}) \tag{18}$$

From equation (18) we can see if MGR is vanishing, heterogeneous object will be a multi-material object, otherwise heterogeneous object is a functionally gradient material object.

2.2.2 FRF&AGS based material evaluation paradigm

From above subsection, we can see that Distance Field based method can only evaluate three compositions and two materials variations. It is inflexible and undesirable. Siu and Tan (Siu (2000)) proposed the 'source-based' scheme to represent any kind of material primitives according to the material feature. We modify this approach into our framework to overcome the drawback of distance field based method. As the computational expense is tremendous when taking a curve surface or model's contour as a feature, it is sensible to fix the feature unmovable when the material grading source is modified. The unmovable feature(s) are called fixed reference feature(s) (FRF), and the movable grading source is called active gradient source (AGS). By using this scheme, the 'source-based' approach can be effectively used in our HO representation framework.

A material vector of 'source-based' scheme in material gradient range can be modified as the Equation 19

$$\boldsymbol{P}_{mi} = \begin{bmatrix} \boldsymbol{m}_1 \\ \boldsymbol{m}_2 \\ \vdots \\ \boldsymbol{m}_m \end{bmatrix} = f(d - d_{-eff}) \times \begin{bmatrix} m_{e1} - m_{s1} \\ m_{e2} - m_{s2} \\ \vdots \\ m_{ej} - m_{sj} \end{bmatrix} + \begin{bmatrix} m_{s1} \\ m_{s2} \\ \vdots \\ m_{sj} \end{bmatrix}, \left\{ \begin{array}{l} m_{sj} \in \boldsymbol{M}_s \\ m_{ej} \in \boldsymbol{M}_e \\ 0 \leq f(d - d_{-eff}) \leq 1 \end{array} \right\} \tag{19}$$

where $\boldsymbol{M}_s$ and $\boldsymbol{M}_e$ are the material vectors in start and end point of composition variations. Above equation is simplified as

$$\boldsymbol{P}_{mi} = f(d - d_{eff})\boldsymbol{S}_m + \boldsymbol{M}_s, \tag{20}$$

where $\boldsymbol{S}_m = \boldsymbol{M}_e - \boldsymbol{M}_s$

Similar with the distance field based method, the geometric and the material space are divided into three areas respectively, denoted as G_{-eff}, G_{eff} and G_{+eff} and B_{-eff}, B_{eff} and B_{+eff} . Also using $G_{B_{-eff}}$, $G_{B_{eff}}$ and $G_{B_{+eff}}$ to denote the composition constitution in material space, FRF&AGS base representation scheme can be also expressed by Equation 18. But in this case, $f(d)$ must be equal to zero, that is $f(d) = 0$, in $B_{-\text{eff}}$, and $f(d) = 1$ in $B_{+\text{eff}}$. With respect to just one material feature and one grading source, the composition is evaluated as follows.

$$G_{B_{-eff}} = \{\boldsymbol{P}_{mi}|\boldsymbol{P}_{ei} \in G_{-eff}\&\boldsymbol{P}_{mi} \in B_{-eff}\&f(d) = 0\} \tag{21}$$

$$G_{B_{eff}} = \{\boldsymbol{P}_{mi}|\boldsymbol{P}_{ei} \in G_{eff}\&\boldsymbol{P}_{mi} \in B_{eff}\} \tag{22}$$

$$G_{B_{+eff}} = \{\boldsymbol{P}_{mi}|\boldsymbol{P}_{ei} \in G_{+eff}\&\boldsymbol{P}_{mi} \in B_{+eff}\&f(d) = 1\} \tag{23}$$

From Equation 19 we can see if d_{eff} equals to zero, heterogeneous object is FGM, otherwise it is MMO. Boolean operators facilitate the set operation in solid modeling. Likewise, we can also define heterogeneous representation Boolean operators. As we take volumetric dataset to represent the heterogeneous object, it is convenient to execute the Boolean operation.

2.2.3 HO representation operators

In order to facilitate the modeling of heterogeneous object, we design some representation operators, such as AGS Add, Delete, Move and HOs Intersection, Union, Difference, Merge, etc.

- Operator for AGSs Union: $\bigcup_g^*$.
  If there is more than one AGS inside the object, we need an operator to unify the different AGSs, denoted as $\bigcup_g^*$. G_1 and G_2 are two AGSs. The union operation is defined as $G = G_1 \bigcup_g^* G_2$, where $G_1 = ((G_{-eff1}, G_{B_{-eff1}}), (G_{eff1}, G_{B_{eff1}}), (G_{+eff1}, G_{B_{+eff1}}))$, and $G_2 = ((G_{-eff2}, G_{B_{-eff2}}), (G_{eff2}, G_{B_{eff2}}), (G_{+eff2}, G_{B_{+eff2}}))$. After union operation, we can get three fields G^3_{-eff}, G^3_{eff} and G^3_{+eff}.

- Operator for AGS Add: $\oplus_{ags}$.
  This operator will add a new AGS feature into the heterogeneous object. It is defined as follows.

$$\oplus_{ags} = \{(G_{-eff}, G_{B_{-eff}}), (G_{eff}, G_{B_{eff}}), (G_{+eff}, G_{B_{+eff}}) | d_{-eff} = l_1,$$
$$d_{eff} = l_2, d_{+eff} = l_3; l_1, l_2, l_3 \in [0, \max(d)]\} \tag{24}$$

- Operator for AGS Delete: $\ominus_{ags}$.
  One existing AGS features can be deleted by this operators which is defined as

$$\ominus_{ags} = \{(G_{-eff}, G_{B_{-eff}}), (G_{eff}, G_{B_{eff}}), (G_{+eff}, G_{B_{+eff}}) | d_{-eff} = 0,$$
$$d_{eff} = 0, d_{+eff} = 0\} \tag{25}$$

- Operator for AGS Move: $\otimes_{ags}$.
  This operator is used to move an AGS to a new position, which can be employed to change the material variation. It is defined as

$$\otimes_{ags} = \{(G_{-eff}, G_{B_{-eff}}), (G_{eff}, G_{B_{eff}}), (G_{+eff}, G_{B_{+eff}}) | d_{-eff} = l_1,$$
$$l_1 \in [0, \max(d)]\} \tag{26}$$

- Operator for HOs intersection: $\overline{\cap}_G$
  H_1 and H_2 are two heterogeneous objects. There are n AGSs in H_1 and H_2, denoted as G_{n1} and G_{n2}. Then, H_1 and H_2 can be represented as $H_1 = (O, G_{n1})$ and $H_2 = (O, G_{n2})$. The intersection operation of these two model is defined as

$$H_1 \overline{\cap}_G H_2 = (O_1 \cap^* O_2), \oplus_{ags}(G_{n1} \cup^* G_{n2}) \tag{27}$$

- Operator for HOs union: $\overline{\cup}_G$.
  This operator is used to unify to two heterogenous objects, H_1 and H_2. The definition is as follows.

$$H_1 \overline{\cup}_G H_2 = \left\{ \begin{array}{l} (O_1 -^* O_2, G_{n1}) \\ (O_1 \cap^* O_2, \oplus_{ags}(G_{n1} \cup^* G_{n2})) \\ (O_2 -^* O_1, G_{n2}) \end{array} \right\} \tag{28}$$

- Operator for HOs difference: $\overline{\ominus}_G$: The difference between two heterogenous objects, H_1 and H_2, is defined as

$$H_1 \overline{\ominus}_G H_2 = \left\{ \begin{array}{l} (O_1 -^* O_2, G_{n1}) \\ (O_2 \cap^* O_1, G_{n2}) \end{array} \right\} \tag{29}$$

- Operator for HOs merge: $\overline{\oplus}_G$
  This operator can be utilized to merge two heterogenous objects, H_1 and H_2. The definition of merge operator is

$$H_1 \overline{\oplus}_G H_2 = (O_1 \cup^* O_2), (G_{n1} \cup^* G_{n2}) \tag{30}$$

2.3 Heterogeneous object visualization

As the volumetric dataset is a discrete representation of an object, the normal is lost in the process of voxelization. Thus, the rendered image of HO is not realistic. However, direct volume rendering (DVR) in scientific visualization is a powerful tool to render volumetric datasets. DVR technique is mainly used in medical imaging, where volume date is available from CT, MRI or PET. DVR is an approximate simulation of the propagation of the light through a colored, semi-transparent gel where the color and opacity are functions of the scalar values in the volume dataset. The DVR algorithms fall into two categories, namely image based method and object based method, according to the ways of voxel projection. In our approach we use a modified ray-casting algorithm to render volumetric dataset of heterogeneous object. Traditionally, volumetric dataset is projected onto an image plane by assigning a color and opacity to each voxel. The standard ray-casting pipeline is showed in Fig.8(a). For the HO volumetric dataset, the color information has been computed to represent the material properties. The modified ray-casting pipeline utilized in HO representation is displayed in Fig. 8(b). Fig. 9 gives some rendered results from different volumetric dataset of heterogeneous object, from which we can see that the images reveal the realistic appearance of 3D object and the transparency by proper rendering parameters.
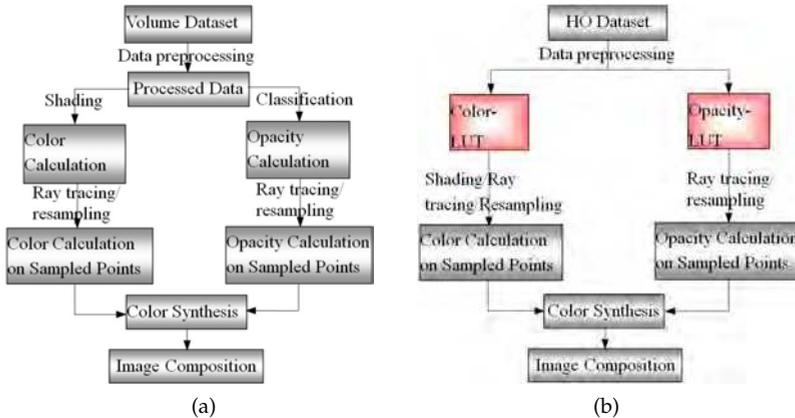
Fig. 8. Standard ray-casting pipeline (left), and Modified ray-casting pipeline (right)
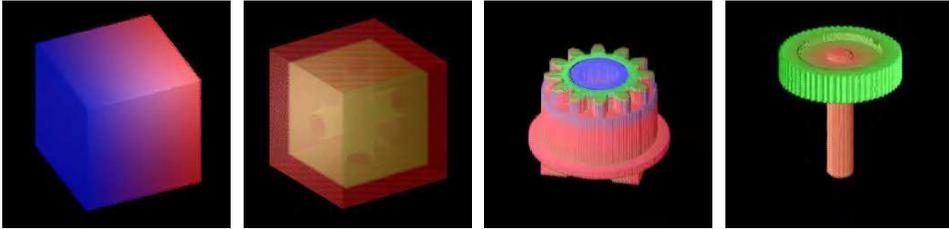
Fig. 9. Ray-casting rendered images of HO

2.4 Heterogeneous object slicing
2.4.1 Geometry model slicing

One of the prerequisites of rapid prototyping manufacturing is slicing the geometry model into 2D contours such that the path planning can be generated (Adams (2004), Mani (1999), Tata (1998)). The geometric slicing algorithm was extensively studied in RP community in the past decades. It can be classified into two categories: slicing polygonal meshes with same thickness or adaptive layers; and direct slicing of CAD models. Jamieson developed the first direct slicing method on the basis of solid modeling core of UG (Jamieson (1995)). Zhao proposed an adaptive direct slicing scheme for CAD models by using ARX (Zhao (2000)). In our system, we use the Weighted Directed Graph (WDG) to recoding the STL model such that the face table, edge table and vertices table can be well established. WDG is a directed graph that has a weight, or numeric value, associated with each edge in order to eliminate the redundancy and facilitate the traversal of STL models. The benefits of WDG are as follows.

1. Only one intersection point is needed for a triangle when calculating the intersection points as we can inherit another intersection point from the facet connectivity in WDG.

2. All the intersection points are connected as a closed loop sequentially and the reordering is not needed.

3. Once the WDG is constructed, it can be reused when the thickness of the slices does not meet the requirement, which is highly efficient.

Taking a tetrahedron as example, shown in Fig. 10(a), we set the triangle facet as the connection node of the graph, and then give numbers the corresponding edges, seeing Fig. 10(b), and set the weights for the common edges of the neighboring faces such that the WDG can be constructed according to this connectivity attributes, show in Fig.10(c). The tetrahedron is unfolded as Fig. 10(b), then we denote the four triangles $\triangle BCD$, $\triangle ABC$, $\triangle ACD$, and $\triangle ABD$ as $Tri0$, $Tri1$, $Tri2$ and Tri3, and the vertices of each triangles as v_0, v_1 and v_2. In Fig. 10(c), the nodes of the WDG are the four triangles denoted as N_0, N_1, N_2 and N_3, and the arrows point to the neighboring triangle facet. The number on each edge is the weight which is the summation of the subscripts. The weights must comply with the following regulations according to the subscript of the vertex notation. $1 + 0 = 1$ for $v_0 v_1$ or $v_1 v_0$, $2 + 0 = 2$ for $v_0 v_2$ or $v_2 v_0$, $1 + 2 = 3$ for $v_1 v_2$ or $v_2 v_1$.

For efficiently traversing the WDG, we construct an adjacency list to store the nodes, shown in Fig.11, and each node is a singly linked list as shown in Fig. 12. In this case, the redundancy of the STL can be get rid of and mitigate the slicing computation.

After WDG is set up, we can traverse it to calculate the intersection point with depth first search (DFS). The traversal starts from node N_0, then to the next unvisited node N_i neighboring N_0. It is serial to process the next node of the graph. A flag is set to 1 for
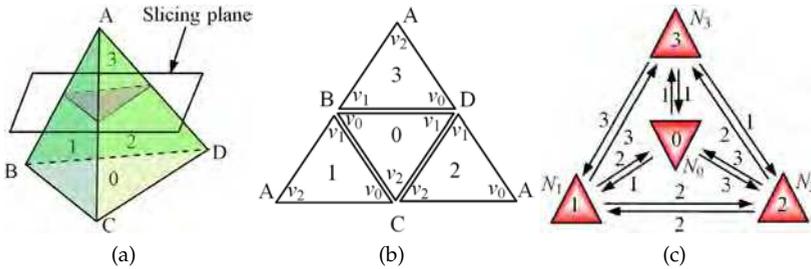
Fig. 10. A solid model: (a)tetrahedron, (b)connectivity of triangles, (c)weighted direct graph of the triangles.
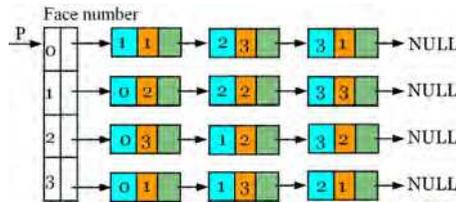

Fig. 11. Adjacency list for triangles.

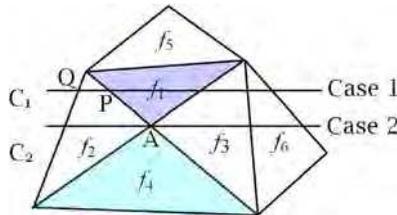Fig. 12. Node of the adjacency list.

Fig. 13. Triangles search schemes.

the traversed triangle as the slicing algorithm need not visit all of the triangles, process the triangles intersected with the slicing plane instead. As illustrated in Fig. 13, when the intersection point, A, locates on the common vertices of face f_1 and f_4 , the flag of face f_2 has been set to 1 if we assume f_1 is the current triangle such that the face f_4 can not be processed. The next face should be f_3. The intersection points can be calculated according to the weights and stored in a doubly linked list. Therefore, it is important to test the intersection between triangle and slicing plane and set a flag to each triangle for efficiently slicing the model. If $flag = 0$ or miss the intersection of the slicing plane with a triangle, the pointer of adjacency list moves to the next triangle to continue the search. When the traversal is accomplished, the slicing contour can be generated. Fig.14 gives an example to validate the geometric slicing algorithm.
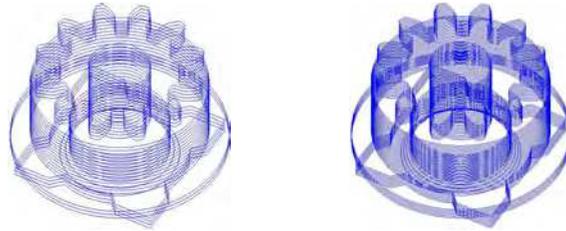
Fig. 14. Geometric slices with different thickness.

2.4.2 Heterogeneous object slicing with geometric constraint

In our HO modeling system, the material information is represented in voxel based model. Because the voxel grid is axis aligned, it is very easy to obtain the material information along x, y, and z axis as illustrated in Fig. 15 (b), (c), and (d) which is obtained by extracting a voxel layer every some continuous layers. Figure (a) is a gear mesh model, and (d) also gives one of the slices along z axis. On the one hand, this volumetric heterogeneous model can not be manufactured directly on account of the discrete representation and the roughness on the boundary. On the other hand, the volumetric model is very memory consuming, and unfeasible to compression, store, and transmission. Therefore, we must transform the volumetric object into portable format to meet the requirement of manufacturing and exchanging. In digital image processing, there are lots of lossless compression techniques to facilitate the data store and transmission.

(a) STL mesh   (b) Slices along x (c) Slices along y (d) Slices along z and one slice show
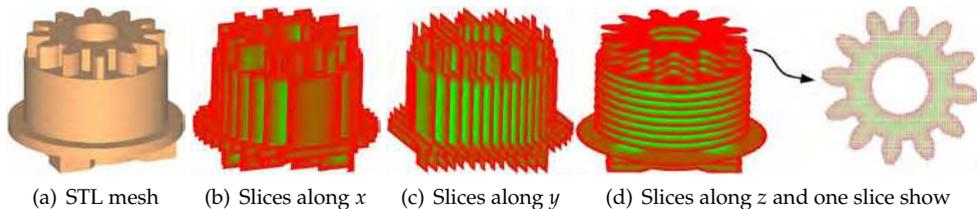
Fig. 15. Material slicing of a heterogeneous object.

As aforementioned, the discrete heterogeneous model can not offer adequate accuracy for fabrication because of the low resolution of voxel layers. For example, Fig. 16(a) is a bitmap liked image formed directly from a voxel layer (b) which is in object space. But the resolution is relatively low. Figure (c) is portion of zoomed voxel layer, from which we can find that the limited resolution can not completely represent the contour boundary. The blue curve is the geometry slice where some of the voxels locate outside the curve boundary. Fig. 16 (d) is representation of the geometric curve and pixels whose color of the dot represents the different attributes demonstrated as the notations in Fig. 16. Figure (e) is an enlarged image of the figure (d) from which we can conclude that the boundary of the solid model can be accurately represented with more interpolations. However, with the increase of the resolution, more and more object pixels become background pixels, the white dots in Fig. 16(d). We must offer a scheme to determine whether a pixel belongs to object or background. In this case, the geometry slicing contours can be used as constraints to restrict the interpolation. It is implemented by image interpolation method to resample the material slicing, which can be

carried out in 2D and 3D. In 2D, it is regarded as interpolation inside an image, and in 3D one or more new images can be constructed with interpolation between neighboring two images.
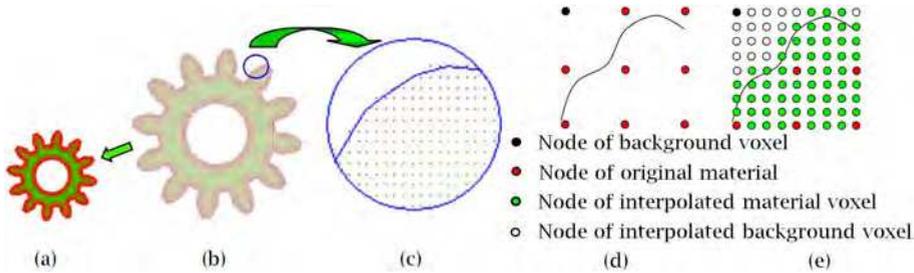

Fig. 16. Original voxel layer image and the interpolation.

Traditionally, the nearest, bilinear, bicubic interpolation methods can be used to improve the image resolution (Gmoldwasser (1988), Maeland (1988)). However, the pixels in new interpolated image can not maintain the information of the original image. Yokoya proposed an image interpolation method based on fractal geometry (Yokoya (1989)), which used the statistical self-similarity between gray levels of neighboring pixels to interpolate. Whereas it is difficult to accurately compute the self-similarity transformation using the traditional fractal scheme. Furthermore, the information can not be well maintained between the interpolated image and the original voxel layer after several times resampling. In our scheme, we employ a method called pixel overlap interpolation (POI) to maintain the material information to the greatest extend, which can keep all of the original pixel values when up-sampling.

The pixels are regarded as rectangular instead of pure points in POI, as shown in Fig. 17(a). When executing the interpolation, the original image is extended like a plastic membrane to the same size with the interpolated image. Thus, the original image can cover all the area of the new image, shown in Fig. 17(b), and Fig. 18(b). The new interpolated image can be obtained by calculating the information percentage of the overlapped rectangles. For example Fig. 17(a), if we contract the image into 3×3 , more pixels cover the overlapped rectangles; the number is 9 pixels at most, see Fig. 17(b). Original image includes 25 pixels, from O_1 to O_{25}. The pixel, N_1 , in new image consists $36\%O_1$, $24\%O_2$, $24\%O_6$, and $16\%O_7$ respectively. That is $N_1 = 0.36O_1 + 0.24(O_2 + O_6) + 0.16O_7$.

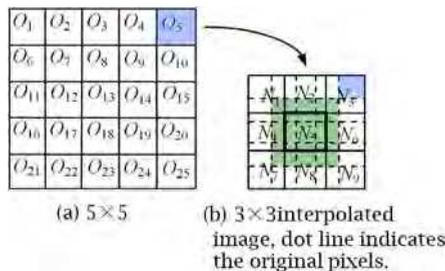Likewise, $N_2 = 0.12(O_2 + O_4) + 0.36O_3 + 0.08(O_7 + O_9) + 0.24O_8$.

(a) 5×5 (b) 3×3 interpolated image, dot line indicates the original pixels.

Fig. 17. 5×5 image contracts into 3×3, the bold box is covered by nine pixels.

If the new image is enlarged, the maximum pixels in the new image are not more than four overlapped pixels, see pixel N_5 in Fig. 9(b). In this case, the new image can be constructed
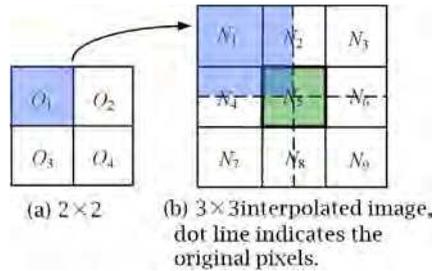
(a) 2×2

(b) 3×3 interpolated image, dot line indicates the original pixels.

Fig. 18. 2 × 2 image is scale into 3 × 3, the bold rectangle is occupied by four pixels.

$N_1 = O_1$, $N_3 = O_2$, $N_7 = O_3$, $N_9 = O_4$, $N_2 = 0.5(O_1 + O_2)$ (so do N_4, N_6, N_8) and $N_5 = 0.25(O_1 + O_2 + O_3 + O_4)$. The advantage of this interpolation method is maintaining the information to the greatest extend which is very suitable to store the heterogeneous material compositions.

As our heterogeneous object is represented as volumetric data set, the material information in the inter-layer can be computed by trilinear interpolation which is a method of multivariate interpolation on a 3D regular gird. It approximates the value of an intermediate point within the local axial rectangular prism linearly. A sample point is on the geometry slice plane and is within a voxel, indicated in Fig. 19. The material information on can be calculated using Equation 31.

$$B = P_0(1 - u)(1 - n)(1 - w) + P_1 u(1 - v)(1 - w) + P_2 v(1 - u)(1 - w+)$$
$$P_3 w(1 - u)(1 - v) + P_4 uv(1 - w) + P_5 uw(1 - v) + P_6(1 - u) + P_7 uvw \qquad (31)$$

where u, v and w are the local coordinates on a voxel grid. They are computed as follows.

$$u = x_b - x_{p0}, \quad v = y_b - y_{p0}, \quad w = z_b - z_{p0} \qquad (32)$$

where (x_b, y_b, z_b) and (x_{p0}, y_{p0}, z_{p0}) are the coordinates of sample point B and voxel vertex P_0.
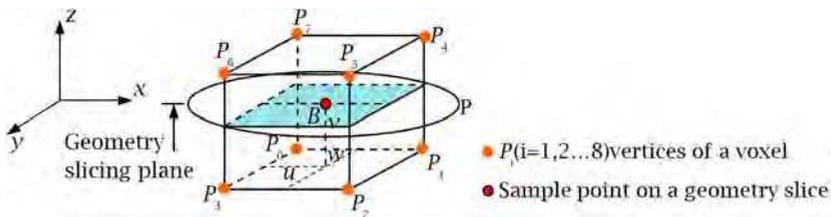
Fig. 19. Sample point and trilinear interpolation

In the above section, we describe the interpolation methods in 2D inner-layer and 3D inter-layer. Now, we can start to construct HO slicing with any desired precision. Fig. 20(a) is a STL model, (b) voxel layers with heterogeneous material and geometric slicing contours, (c) a new view of (b) along x axis, and (d) a zoomed view of a portion of (c), and (e) one of layers of (d) viewed along z axis. From figure (c)(d) and (e), we can conclude that our method implies two stages. Firstly, when the voxel layer resolution along the direction of geometry slicing is less than that of geometric slices, trilinear interpolation method is utilized to get the heterogeneous material information on the geometric planes such that the thickness of the voxel layer is same as the geometric slices. Secondly, when the resolution and shape accuracy

of the voxel layer can not satisfy the requirement of manufacturing, the voxel layer should be resampled by using POI method on the geometry slice plane.
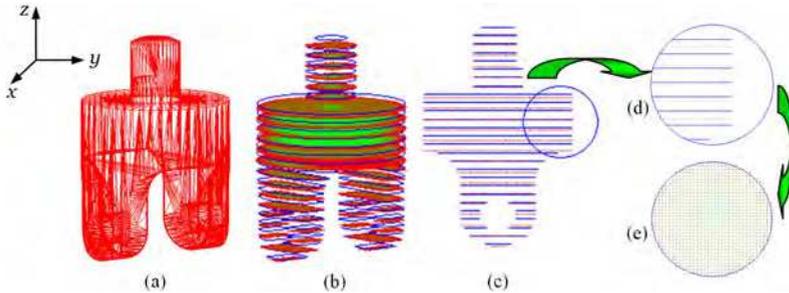

Fig. 20. HMP slicing and geometric slicing, (a) geometric model, (b) geometric slices and voxel layers, (c)(d)(e) different view of the geometric slices and voxel layers.

The most complicated part of HO slicing is the resampling operation on the voxel layers as we have to consider the boundary to grantee the accuracy and smoothness. The initial heterogeneous material layer is denoted as I_0, I_1 is the first sampled image, then next $I_2, \cdots, I_k$. If the resolution can satisfy the requirement of manufacturing, the sampling procedure can be stopped, illustrated in Fig. 21. Figure (d) is the third resampling from which we can find that the pixels outside the geometric boundary have not any contribution to the shape representation. The eight pixels under the shadow is an example in Figure (d). These pixels can be changed into background pixels. In practice, a threshold d is set to filter the non-contributive pixels. If the distance from the pixel to the geometric contour is greater than d, this pixel is set to background pixel; otherwise, this pixel is maintained as object pixel. In this scheme, the accuracy of the final HO slice is totally decided by the number of resampling, demonstrated in Fig.21(e).
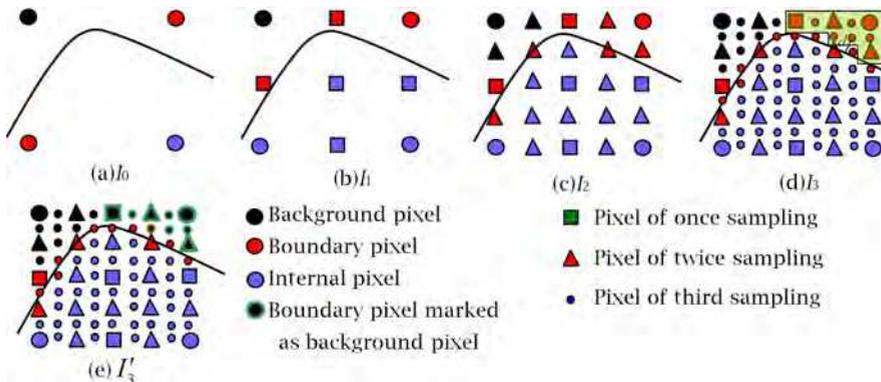
Fig. 21. Iteratively sampling in 2D, (a) is the original image, (b)(c)(d) are three times resampling (d) is the final high resolution image after eliminating the non-contributive pixels.

3. Modeling examples and analysis

In this section, three examples are presented to show the validity of our approaches to model heterogeneous object. Firstly we model a multi-material object model. A STL mesh and its

voxelized model are shown in Fig. 22. This component is constituted by three materials, e.g. $m = 3$. We select the y axis as the reference feature from which the distance map is computed. Then, we define $d_{-eff} = 21.17$, $d_{eff} = 0$ and $d_{+eff} = 8.83$, and the material distribution function is $f(d) = 0$, e.g. the gradient field is none in MMO model. Through Equation (14) to calculate M_s and M_e, so we can get $M_s = [0.1, 0.7, 0.2]$ and $M_e = [0.7, 0.1, 0.2]$. Finally, the material composition can be obtained by equation (15) and (17). The created multi-material object is illustrated in Fig. 23 (a) and (b). (c) is one slice intersected from a plane perpendicular z axis. To analyze the material distribution in object, we firstly plot the material distribution on the slice shown in figure (d) from which we can found there is abrupt change on the boundary of the part. Then we extract two sample lines, S_1 and S_2, on the slice, which is demonstrated in figure (c). The plotted curves show that there are three materials on the line and the material distribution is accordance with the principle of MMO model.
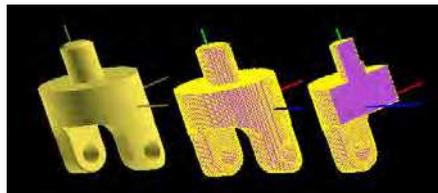
Fig. 22. A STL triangle mesh and voxelized model.
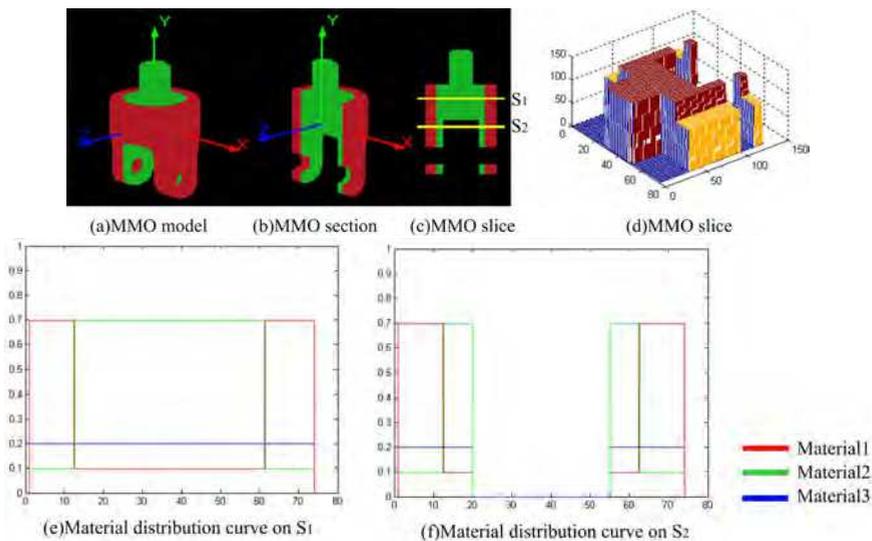
Fig. 23. Multi-material object and material distribution on sampling curves.

The second example is a shaft STL mesh model with demiension $(150, 55, 55)$, see Fig. 24 (a). The volumetric representation is illustrated as (b) with resolution of $(256, 97, 97)$ voxels. This component consists of three composition primitives. The contour of the model is chosen as reference feature to calculate distance map. The modeling parameters are as follows, $m = 3$, $d_{-eff} = 0$, $d_{eff} = 20.416$ and $d_{+eff} = 0$. A sinusoid function, $f(d) = \sin(0.1538d)$, is taken as material distribution function. The material vectors at the start and end point are

$M_s = [0, 0, 1.0]$ and $M_e = [0, 0.2, 0.8]$. Equations (21), (22) and (23) are utilized to evaluate the compositions. The final results are presented in Fig. 24 (c) and (d).
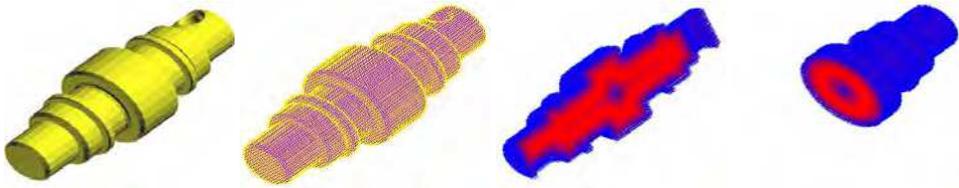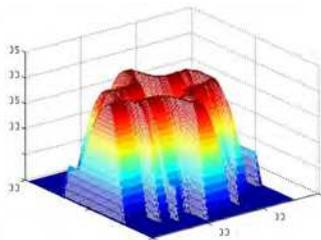

Fig. 24. HO representation of a shaft

In Fig.25, we extract the slices of the HO along axis and radius, showed in (b) and (c). Figures (d) (e) and (f) (g) are the corresponding material spatial distribution on slices (b) and (c), from which we can see that the material is evaluated in accordance with material distribution function. Furthermore, the HO slices are sampled on three orthogonal lines through the center
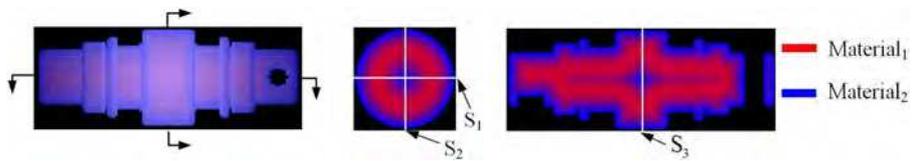
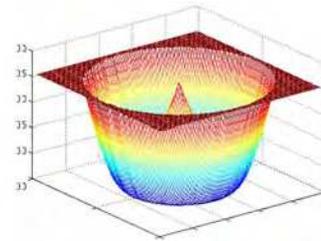| (a) Heterogeneous object | (b) Slice and sample liens on radial direction | (c) Slice and sample line in axis direction |

| (d) Material1 distribution on slicing(b) | (e) Material2 distribution on slicing(b) |

| (f) Material1 distribution on slicing(c) | (g) Material2 distribution on slicing(c) |
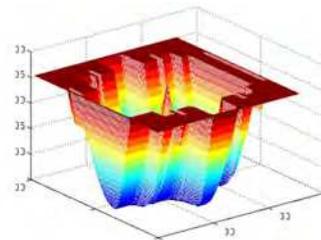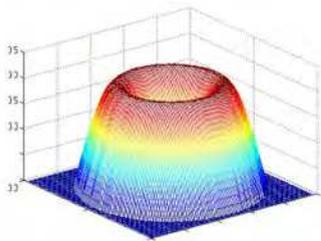
Fig. 25. HO and material distribution on slices

of the shaft, shown in (b) and (c) lines S_1, S_2 and S_3. The curve shown in Fig.26 (a) is the material distribution function (MDF) graph according to a given distance map Figs.26 (b) (c) and (d) show the material distribution (MD) on the sample lines, from which we can see that the results from our system is almost same as the theoretical model.
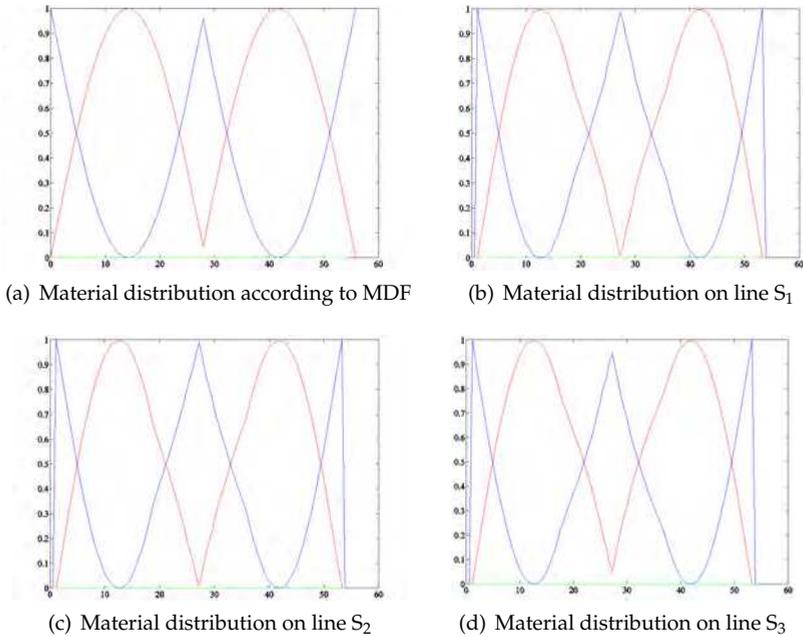

(a) Material distribution according to MDF   (b) Material distribution on line S_1

(c) Material distribution on line S_2 (d) Material distribution on line S_3

Fig. 26. Material analysis on sample lines

The third example is a compound modeling procedure to create a heterogenous object. Fig. 27 (a) is a gear model which is composed of three materials. Then, the z axis and bottom plane are selected as fixed reference features (FRF). Fig. 27 (b) is the voxelized model. Firstly, we use the add operator to create a HO model as

$$\oplus_{\text{ags}} G_1 = \Big\{ (G_{-eff1}, G_{B_{-eff1}}), (G_{eff1}, G_{B_{eff1}}), (G_{+eff1}, G_{B_{+eff1}})| $$
$$d_{-eff1} = 9.54, d_{eff1} = 9.54, d_{+eff1} = 20.37 \Big\}$$

The material distribution function is a linear function, $f(d) = 0.105d$. The material vector at start and end point is $\boldsymbol{M}_s = [0, 0.9, 0.1]$ and $\boldsymbol{M}_e = [0.9, 0, 0.1]$. The created FGM model is illustrated in Fig. 27 (c) and (d). Next, we take the bottom plane as the FRF, and add the second AGS as

$$\oplus_{\text{ags}} G_2 = \Big\{ (G_{-eff2}, G_{B_{-eff2}}), (G_{eff2}, G_{B_{eff2}}), (G_{+eff2}, G_{B_{+eff2}})| $$
$$d_{-eff1} = 3.18, d_{eff1} = 22.26, d_{+eff1} = 14.01 \Big\}$$

Material distribution is also defined as $f(d) = 0.105d$. The start and end material vectors are $M_s = [0, 0.9, 0.1]$ and $M_e = [0.9, 0, 0.1]$. The created FGM model is demonstrated in Fig. 27 (e) and (f). Then, we use the AGS union operator to unify AGS1 and AGS2 to obtain a HO model with several fixed reference features, illustrated in Fig. 27 (g) and (h). If we consider Fig.27 (f) and (h) are two different heterogenous objects, we can use HOs union operator to unify these two HOs, shown in (m) and (n).
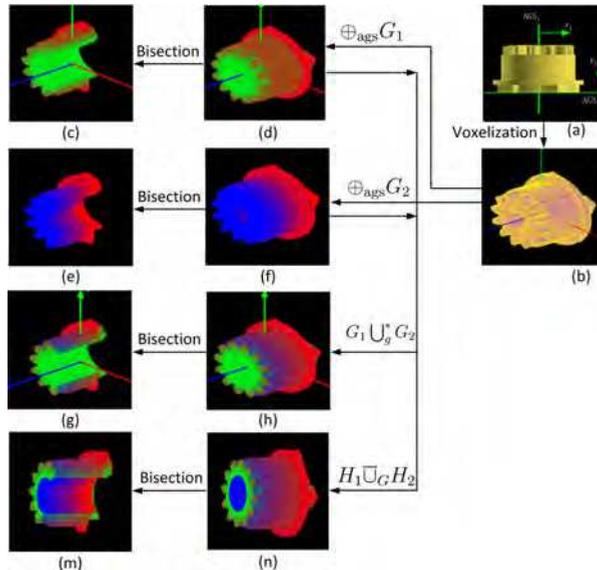

Fig. 27. A compound heterogenous object modeling with different fixed reference features and two AGSs, (a) and (b) are mesh model and voxelized model, (d)(c) are a HO with z axis FRF, (f)(e) are the HO with FRF of bottom plane, (h)(g) are the HO created with AGSs union operator, (n)(m) are the heterogeneous object obtained with HOs union operator.

To analyze the material distribution inside the object, we slice the HO in Fig. 27 (d), (f), (h) and (n) along radial and axial to get the layers illustrated in Fig. 28 (a), (c), (e), (g) and (m). Then we plot the material primitives of each slice to show the distribution in space. The 3D profile of material distribution is revealed in figure (b), (d), (f), (h) and (n). From these figures we can see that the material distribution strictly following the linear property of design function. We also should notice the difference of figure (g) and (m) or (h) and (n), which gives the difference between AGSs union and HOs union.

Fig.29 is an example of heterogeneous object slicing and resampling with geometric contour constraint, where (a) is STL mesh model, (b) the corresponding voxel model, (c) heterogeneous object and geometric slices, thickness of slice is 0.15mm and the number of layers is 117. The amount of voxel layer along z axis is 73. (d) is a hatched view of the heterogeneous object. (e) is a layer of geometric slice contour and a voxel layer with same height. Figure (f) is an image constructed from the voxel laryer directly but with low resolution, only 80×124. Then the image is resampled four times using aforementioned technique, the resolution can achieve 320×496. After the invalid pixels are abandoned, we obtain an image with high resolution and clear boundary exactly with the corresponding geometric contour, for clearness only three portions of the image are displayed in figures (g), (h) and (m). Figure (n) is an enlarged part of
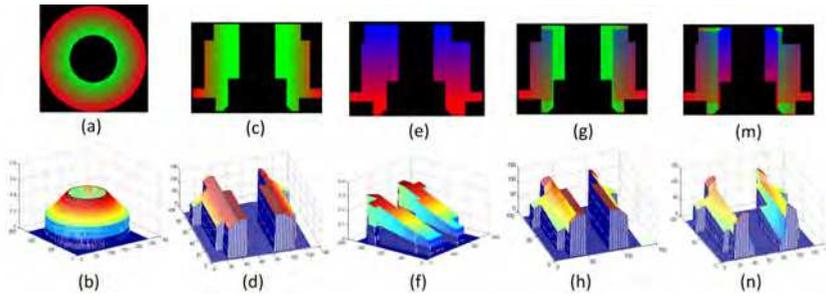
Fig. 28. HO slices in different directions and 3D profile of material distribution in spece.

image (f) using simple interpolation scheme without geometry constraint, from which we can find that the edge of the image is very blurry. We can also use the geometric contours produced by direct slicing or adaptive slicing algorithm as constraints to reconstruct the HO slices. In this case, the accuracy of HO slices is determined completely by the resample resolution. It is clear that we can theoretically construct accurate slices with heterogeneous information exactly as long as the resample resolution is high enough. However, it will increase the computational and storage cost. It is unnecessary to resample the material voxel layer to extremely high resolution. As long as the accuracy of the layers satisfies the manufacturing requirement, it should be stopped. The slices can be employed to produce the path planning using halftone or other methods.
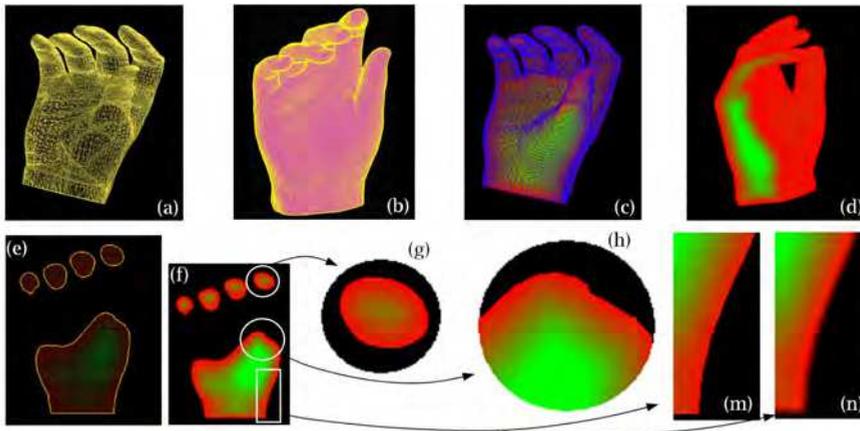

Fig. 29. An example of MRGC, (e) a layer of volume dataset and a geometrical contour, (f) a HO slice constructed from (e) directly, (g)-(m) are three parts of the enlarged image with clear boundaries, (n) is an enlarged part of image (f) using simple interpolation scheme without geometry constraint.

4. Conclusion and future work

In this chapter, we present a prototype system of heterogeneous object modeling independent of any commercial software packages. This system offers a pine-line from CAD model to 2D slices with heterogeneous material. Our approach increases the flexibility of heterogeneous

object modeling with the volumetric dataset structure but not lose the accuracy of HO slices. Moreover, slices of the heterogeneous object can be easily generated and the HO volumetric dataset can be visualized with ray casting method to show the specific material distribution within an object. Our heterogeneous object slicing scheme combines the geometric slice contours and gradient material information. The geometric contours are taken as constraints to resample the heterogeneous information through pixel overlap interpolation and trilinear interpolation to avoid blur on the boundary owing to simple interpolation scheme. The novelty of our method is that it can maintain the original heterogeneous material information as much as possible and the slicing boundary as accurate as possible. The output of our method is images with lossless compression property, which is most suitable for 3D printing technology to manufacture the heterogeneous object through digital half-toning method.

Although mesh model is utilized to represent the geometry of an object in our HO design approach, other 3D file format can be easily integrated into our system as the direct slicing algorithm can generate the slices which do not influence the representation of the material variations. The development of this system can facilitate the design, visualization and fabrication of heterogeneous object. Examples demonstrate the effectiveness of the heterogeneous object modeling system.

However, most of the work of heterogeneous object modeling focuses on the representation of material variations in CAD model. But the most important issue of HO design is the consideration of physical constraints, such as temperature, pressure, stress, friction or load, etc. This is the most difficult issue to design a heterogenous object. Qian et. al. used the conception of physics based heterogenous object design (Qian (2003)). But only the Young's modulus was considered in their design procedure. How to unify all the design parameters into one framework of HO modeling is one of future research directions. We claim that one solution to this problem is task oriented method, which means that for specific issue we can use some specific methods to design heterogenous material variations with consideration of physical constraints. Qian et.al. used this idea to design a turbine blade where the required aerodynamic and mechanical performances were incorporated into the design pipeline (Qian (2003b)).

5. Acknowledgement

6. References

ZCorporation. (2005).*Multi-color object*, `www.zcorp.com`.

Binnard, M.B. (1999), Design by composition for rapid prototyping. *PhD thesis*, Stanford University, 1999.

Fessler, J., Nickel, A., Link, G., et al. Functional gradient metallic prototypes through shape deposition manufacturing. *Proceedings of Symposium on Solid Freeform Fabrication*, University of Texas at Austin, Austin, Texas, August 1997, 521-528.

Jakubenas, K.J., Sanchez, J.M., Marcus, H.L., Multiple material solid free-from fabrication by selective area laser deposition. *Material & Design*, 1998, 19: 11-18.

Ensz, M. T., Griffith, M. L., and Reckaway, D. E., (2002), Critical issues for functionally graded material deposition by laser engineered net shaping (LENS), *Proceedings of the 2002 MPIF Laser Metal Deposition Conference*, San Antonio, TX.

Marsan, A. L.; Allen, S.; Kulkarni, P.; Dutta, D. (1997). An integrated software system for process planning for layered manufacturing, *Proceedings of the Solid Freeform Fabrication Symposium*, 1997, 661-668.

Siu, Y. K.; Tan, S. T.(2002). Modeling the material grading and structures of heterogeneous objects for layered manufacturing, *Computer-Aided Design*, 34(10), 2002, 705-716.

Siu, Y. K.; Tan, S. T. (2002). Slicing and contours generation for fabricating heterogeneous objects, *Proceedings of the Geometric Modeling and Processing-Theory and Applications (GMP'02)*, 2002, 219-225.

Kumar, V.; Dutta, D. (1997). An approach to modeling multi-material objects, *Proceedings of the fourth ACM symposium on Solid modeling and applications*, 1997, 336-345.

Kumar, V.; Dutta, D (1998). An approach to modeling & representation of heterogeneous objects, *Journal of Mechanical Design*, 120, 1998, 659-667.

Jackson, T.R. (2000). Analysis of functionally graded material object representation methods, *PhD thesis*, Massachusetts Institute of Technology (MIT), Cambridge, MA, January 2000.

Liu, H. (2000). Algorithms for design and interrogation of functionally graded material solids, *Master's thesis*, Massachusetts Institute of Technology, Cambridge, MA, January 2000.

Chiu, W. K.; Tan S.T. (2000). Multiple material objects: from CAD representation to data format for rapid prototyping. *Computer-Aided Design*, 32, 2000, 707-717.

Marsan, A.; Dutta, D. (1998). On the application of tensor product solids in heterogeneous solid modeling. *Proceedings of 1998 ASME Design Engineering Conferences*, Atlanta, Georgia, Sep. 1998, 1-9.

Siu, Y. K.; Tan, S. T. (2002). 'Source-based' heterogeneous solid modeling. *Computer-Aided Design*, 34(1), 2002, 41-55.

Qian, X.; Dutta, D. (2003). Physics-based modeling for heterogeneous object, *Journal of Mechanical Design*, 125(3), 2003, 416-427.

Qian, X.; Dutta, D. (2004), Feature-based design for heterogeneous objects, *Computer-Aided Design*, 36(12), 2004, 1263-1278.

Kou, X.Y.; Tan, S. T. (2005). A hierarchical representation for heterogeneous object modeling, *Computer-Aided Design*, 37(3), 2005, 307-319.

Samanta, K.; Koc, B. (2004). Heterogeneous object design with material feature blending, *Computer-Aided Design and Applications*, 1(1-4), 2004, 429-437.

Zhang, X. F.; Subbarayan, G. (2004). A Constructive Approach for Heterogeneous Material Modeling and Analysis, *Computer-Aided Design and Applications*, 1(1-4), 2004, 171 -178.

Wu, X.J.; Liu, W. J.; Wang, M. Y. (2007). Modeling Heterogeneous Objects in CAD, *Computer-Aided Design & Applications*, 4(6), 2007, 731-740.

Wang, M. Y.; Wang, X. M. (2004), A level-set based variational method for design and optimization of heterogeneous objects, *Computer-Aided Design*, 37(3), 2004, 321-337.

Biswas, A.; Shapiro, V.; Tsukanov, I. (2004). Heterogeneous material modeling with distance fields, *Computer Aided Geometric Design*, 21(3), 2004, 215-242.

Hu, Y.; Blouin, V.Y.; Fadel, G.M. (2008). Design for Manufacturing of 3D Heterogeneous Objects With Processing Time Consideration. *Journal of Mechanical Design*, 130(3).

Kou, X.Y., Tan, S.T.(2006). Data Structure and Algorithms for Virtual Prototyping of Heterogeneous Objects, *Computer-Aided Design & Application*, 2006, 3(1-4): 59-67.

Tsukanov, I.; Sapiro, V. (2003). Meshfree modeling and analysis of physical fields in heterogeneous media, *Advances in Computational Mathematics*, 2003. `http://sal-cnc.me.wisc.edu/publications/analysis_in_fgm/analysis_in_fgm.pdf`

Liu, H.; Maekawa; T.;, Patrikalakis, N.M.; Sachs, E.M.; Cho, W. (2004). Methods for feature-based design of heterogeneous solids. *Computer-Aided Design*, 2004, 36, pp 1141-1159.

Kou, X.Y.; Tan, S.T. (2004), A hierarchical representation for heterogeneous object modeling, *Computer-Aided Design*, 2004.

Kou, X.Y.; Tan, S.T. (2007), Heterogeneous object modeling: A review. *Computer-Aided Design*. 2007, 39(4): 284-301.

Kaufman, A.; Cohen,D.; Yagel,R. (1993). Volume Graphics. *IEEE Computer*, 1993, 26(7), 51-64.

Chandru,V.; Manohar, S.; Prakash, C.E. (1995), Voxel-based modeling for layered manufacturing. *IEEE Computer Graphics and Applications*, 1995, 15(6), 42-47.

Kaufman, A.; Shimony, E. (1987). 3D scan conversion of polygons, *Proceedings of EUROGRAPHICS'87 Conference*, Amsterdam, The Netherlands, August 1987, 197-208.

Jones, M. W.; Satherley, R. (2000). Voxelisation: Modeling for Volume Graphics. In B. Girod, G. Greiner, H. Niemann, H. -P. Seidel (eds.), *Vision, Modeling, and Visualization*, 2000, IOS Press, 319-326.

Kong, T.Y.; Rosenfeld, A. (1989), Digital Topology: Introduction and Survey. *Computer Vision, Graphics, and Image Processing Archive*, 1989, 48(3), 357-393.

Huang, J.; Yagel, R.; Fillipov, V.; Kurzion, Y. (1998). An accurate method to voxelize polygonal meshes, *IEEE Volume Visualization*, October, 1998, Chapel Hill, North Carolina, USA, 119-126.

Adams, S.H.; Yang, M.Y. (2004). A study on a generalized parametric interpolator with real-time jerk-limited acceleration, *Computer-Aided Design*, 36(1), 2004, 27-36.

Mani, K.; Kulkarni, P.; Dutta, D. (1999). Region-based adaptive slicing. Computer Aided Design, 31(5), 1999, 317-333.

Tata, K.; Fadel, G.; Bagchi, A.; Aziz, N. (1998). Efficient slicing for layered manufacturing. *Rapid Prototyping Journal*, 4(4), 1998, 151-167.

Jamieson, R.; Hacker, H. (1995). Direct slicing of CAD models for rapid prototyping. *Rapid Prototyping Journal*, 3(1), 1995, 12-19.

Zhao, Z.; Laperriere, L. (2000). Adaptive direct slicing of the solid model for rapid prototyping. *International Journal of Production Research*, 38(3), 2000, 89-98.

Gmoldwasser, S. M.; Reymolds, R. A.; Talton, D. A.; (1988). Techniques for the rapid display and manipulation of 3-D biomedical data. *Comut. Med. Imag. Graph.*, 12(1), 1988, 1-24.

Maeland, E. (1988). On the comparison of interpolation methods. *IEEE Transactions on Medical Imaging*, 7(3), 1988, 213-217.

Yokoya, N.; Yamamoto, K. (1989). Fractal-based analysis and interpolation of 3D natural surface shapes and their application to terrain modeling. *Computer Vision, Graphics and Image Processing*, 46(3), 1989, 284-302.

Qian, X.P.; Dutta, D. (2003). Design of heterogeneous turbine blade. *Computer-Aided Design*, 2003, 35, 319-329.

Rapid Prototyping Technology - Principles and Functional Requirements

Edited by Dr. M. Hoque

Modern engineering often deals with customized design that requires easy, low-cost and rapid fabrication. Rapid prototyping (RP) is a popular technology that enables quick and easy fabrication of customized forms/objects directly from computer aided design (CAD) model. The needs for quick product development, decreased time to market, and highly customized and low quantity parts are driving the demand for RP technology. Today, RP technology also known as solid freeform fabrication (SFF) or desktop manufacturing (DM) or layer manufacturing (LM) is regarded as an efficient tool to bring the product concept into the product realization rapidly. Though all the RP technologies are additive they are still different from each other in the way of building layers and/or nature of building materials. This book delivers up-to-date information about RP technology focusing on the overview of the principles, functional requirements, design constraints etc. of specific technology.

How to reference

In order to correctly reference this scholarly work, feel free to copy and paste the following:

INTECH
open science | open minds