Chapter 4

4. Edit Physical Model

4.1 Material Qualities

In chapter 3, the properties for the prototyping material are outlined. One requirement of the material is that it can be manipulated by carving, sanding, machining, and joining. Each of these requirements have been met.

4.1.1 Carving

Carving is the easiest requirement to satisfy. Most materials can be cut and carved. This prototyping material needs to yield to simple hand carving pressures. The material behaves much like hard soap when carving; although, it does not begin to melt in your hand like soap. Deep cuts remove shavings such as those you would see when carving wood while light cuts with a blade remove only small amounts.

Since most of the structural strength of the material comes from the wax, no appreciable damage is done to the blade during carving. This is significant because glass is abrasive and if there were glass bonds between individual powder elements then most tools would wear out quickly.

4.1.2 Sanding

Sanding is much like sanding wood. The feel is exactly the same. There is a perceived difference in the way the material actually comes off. Since the designer is sanding wax, wax gets in the sandpaper and a thin film of wax about as deep as the grit of the sand paper permanently stays there. Nevertheless, the function of the sandpaper has not altered. Vigorous sanding is not recommended. If a great deal of material needs to be removed, then carving should be employed and sand only as a finishing touch.

Since the material is malleable with hand pressure, finish sanding can often be accomplished simply by rubbing finger tips over an area. This actually gives a better finish than using sand paper.

4.1.3 Joining

There are two different types of joining. They differ in the amount of material being added to the existing object. Small amounts can be added to an object to help cover up imperfections such as in finish sanding or filling in nicks and holes. This can be accomplished simply by taking shavings of the material and working them into the part by hand. The added material quickly joins the object and disappears like lotion on dry skin by simply rubbing gently using fingertips.

The fact that material can be added simply and easily is a very powerful tool and major improvement over existing practice and therefore needs special mention. In order to add small amounts of material to wood, RenShape, or dense foam, a designer uses a compound called Bondo. It is a sticky paste which hardens after the addition of a catalyst. It is messy and difficult to work with. In an iterative process where a designer would try to find just the right shape by adding and subtracting material over and over again, Bondo is a terrible material but is still the premiere in model shops. The bees wax and glass compound make the iterative process of adding and subtracting easy.

The other type of joining is major pieces being “glued” together. Because of the wax, the material has some tackiness. Simply by pressing two pieces together they will stay, but their bond is very weak. In order to make a much better bond, a heat gun directed toward the area of contact melts the wax slightly and makes for a perfect glue. The bond is just as good as anywhere else in the part. To hide the joint, simply do as mentioned above and rub in a small amount of material and the joint will become perfect visually as well as structurally.

4.1.4 Machining

Sometimes major work needs to be done to an object. Most operations that can be done with a machine tool such as a Boolean operation can probably better be done in the computer environment rather than on the physical object. Nevertheless, if it necessary the prototyping material can also be machined.

The material has internal strength to handle machining operations. If something did go wrong and the part cracked or broke, it could always be “glued” back together with no harm as described in the previous section.

4.2 Survey

A formal controlled survey of practicing model makers over a substantial time period would have been a nice addition to this thesis. Unfortunately, this was not done. However, in an informal, non controlled survey of people who had done some modeling of objects before, the material described in this thesis had a favorable first impression. Initial reactions were typically skeptical. After describing and demonstrating some of the characteristics of the material, however, opinions changed and continued to improve as the person being surveyed worked with the material.

Chapter 5

5. Reverse Engineering

5.1 Method

After the designer is satisfied with the changes to the physical model, the part must be reverse engineered. This requires digitization and is accomplished by decomposing the part layer-by-layer; basically the reverse of how it was created in the first place. Figure 5-1 demonstrates this process.

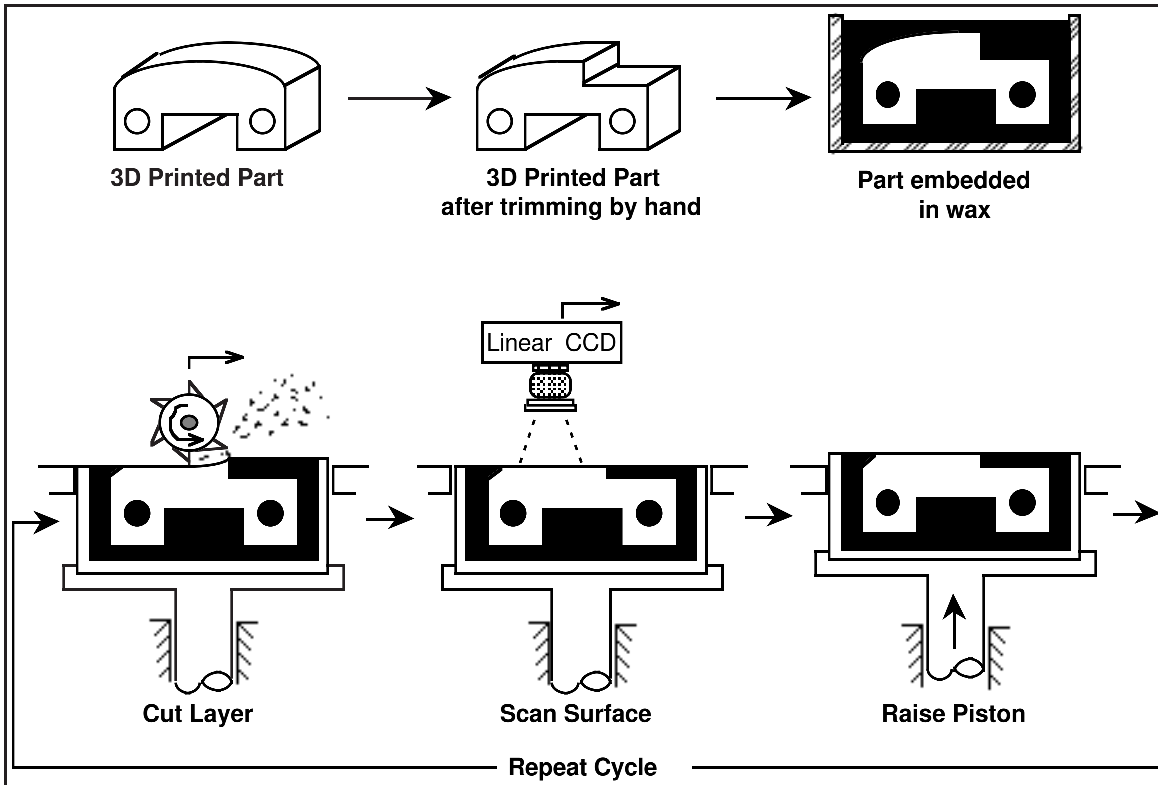


Figure 5-1: Digitizing Sequence of Operations (Deconstruction)

First, the physical model is embedded in a block of material which acts as a physical support and provides optical contrast to the model. A layer of this block of material is shaved off using a simple form of milling. After each layer is removed, an optical scanner captures an image of that layer, based on the optical contrast between the part and the support material. Milling and scanning are repeated until the entire part is imaged.

This approach has several key benefits. First, since the part is decomposed on a layer-by-layer basis and since it is held in place by a support material, any geometry can be imaged. The full set of data is obtained from one “set-up” with no need to fuse the data from multiple views. Further, the data is delivered in a Cartesian coordinate system, permitting easy analysis.

The single disadvantage of this approach is that it is destructive. However, the model is only digitized after all modifications have been made. Since the next stage involves working with the electronic model, there is no need for the physical model after it is digitized. Should a need arise, a new model can always be printed from the electronic model.

5.2 Embedding Material

Finding the right embedding material is a balancing act between its properties and that of the prototyping material. Here two materials are chosen which best fit the required properties. One system is used in the current research and another is recommended as the material for future continuation of the research.

5.2.1 Requirements

The following are the required characteristics of the embedding material.

- Lower melting temperature than the prototyping material. This helps the object maintain its geometry during embedding.
- Does not crack during milling. Cracks and chips show up in the scans as lines and speckles and are difficult to filter out digitally.
- Small change in specific volume from liquid to solid. This insures that the embedding material completely encases the object and does not leave holes or depressions.
- Low viscosity so that it will fill all cavities.

- High visual contrast from prototyping material for better clarity in scanning.
- Opaque so that embedded features remain hidden until cut.
- Fast solidification time.
- Inexpensive.

Many of these characteristics were defined before work began and a few such as the cracking became important only after experimentation.

5.2.2 Experimentation and Analysis

Ten material combinations ranging from waxes to polyester resins and epoxy were tried with varying levels of success. Each is described below.

5.2.2.1 Epoxy and Resin

Clear Epoxy – Obviously epoxy violates some of the most important required material properties and can not be used as an embedding material but it does reveal some important factors. First of all, epoxy at room temperature is viscous and traps many bubbles in the embedding. It also takes a long time (~24 hr.) to solidify. Both of these negative properties can be improved dramatically by raising its temperature since epoxy solidifies in an exothermic reaction. A few degrees is all that is necessary. Still the most glaring problem with clear epoxy is that it is translucent. A scanning device can see a complete silhouette of the entire embedded part and not just the cut surface. With careful digital filtering, the part outline on the cut surface can be found roughly but it is not a good image.

Black Epoxy – I will lump black epoxy and epoxy with a carbon black dispersion in the same category because they had similar results. There are many applications which use black epoxy such as encasing the components of a circuit board for protection. As described above in the clear epoxy section, the viscosity and solidification time can be significantly lowered by slightly raising the epoxy temperature. Also, with the black dispersion, it becomes opaque and should be a good contrast to the part. Unfortunately, when the epoxy is milled, instead of the expected deep dark black, the surface is locally very rough and scatters light yielding a light gray finish.

Polyester Casting Resin with Black Die – This is the other experimental non-wax material. The resin has similar properties to epoxy and reacts the same to temperature. The black die provided by the manufacturer creates a slightly translucent black finished product. Unfortunately, the curing time still took many hours even at elevated temperature.

5.2.2.2 Wax

A host of waxes were tried because of their quick solidification time and also because they are non-caustic.

Candelilla and Carunaba – These are waxes that can be used only with higher melting temperature prototyping materials. Even then they are not good because they are both still quite viscous in their molten state. Also, they are both very brittle. They chip and crack easily during milling therefore are not good embedding materials.

Bees Wax with Carbon Black Dispersion – This wax has most required properties except when it is solidified. Beeswax is a very ductile and tacky material. While that may be good for a prototyping material it is poor as an embedding material because it gums up the cutting tools.

Black Wax Paint – This is a purified beeswax with a black resin and well qualifies for the opaque and high contrast material properties. But, because it is sold as a paint, it is unfortunately quite expensive. Even though it is more suitable as an embedding material than regular bees wax, the expense eliminates it.

Tektronix Black Wax – This is another suitable material but because it is a proprietary peripheral to a commercial product, the expense makes it impossible to use. (The reason that it could be used as part of the prototyping material is that it is simply a binder and not a major portion of the object volume.)

Machinable Wax – This is a prototyping wax used for machining. It is designed to be machined and would be perfect in that respect. However, its melting temperature is significantly higher than all other waxes mentioned. Furthermore, in its molten state, it is very viscous making it unsuitable.

Investment Casting Wax – This wax in its natural state is blue. It can be used either in its natural color or with a carbon black dispersion as described in the Two or Three Colors section of digital registration (section 5.5.2.2 p. 55). This is the only wax with small change in specific volume from liquid to solid. It also is not viscous in the molten state. Its one drawback is that its melting temperature is higher than that of the chosen prototyping material's.

Paraffin with Carbon Black – Paraffin has a viscosity near that of water and has the lowest melting temperature of all the waxes. The carbon black gives it good contrast with the part. It also solidifies within minutes after pouring. The one drawback is that its specific volume change between liquid and solid state is very high. Paraffin is used for candle making and as the candle is cooling, the candlemaker must refill the mold 3-4 times to replace the loss in material volume.

5.2.3 Results

As mentioned at the beginning of this section, two embedding materials are recommended for use. One for use in current research and one for use in future work. The recommended embedding material for use right now is paraffin with carbon black. The exact formula for compound is 20-1-1 by weight: paraffin, carbon black, and stearine. Stearine is a common candle making additive which makes the wax slightly harder and opaque. Suggested use for the mixture is explained in the next section.

When a new 3D printing machine is created with the capabilities of printing a higher melting temperature prototyping material, then the blue investment casting wax with a carbon black dispersion is recommended as the new embedding material because it satisfies all required characteristics.

5.3 Scanning

Since a scanning machine does not currently exist, a procedure is created to mimic the functionality of a machine. Even though there are many manual steps in this procedure, most of it could conceivably be automated.

5.3.1 Setup

After a physical object is altered to a desired shape, it must first be embedded in wax. A container for the part and embedding material is created first from a base of 1/4" aluminum faced parallel on both sides with securing holes drilled every half inch shown in Figure 5-2. The walls for the container are made with aluminum foil which leaves enough room for the part and two reference marks shown in Figure 5-3.

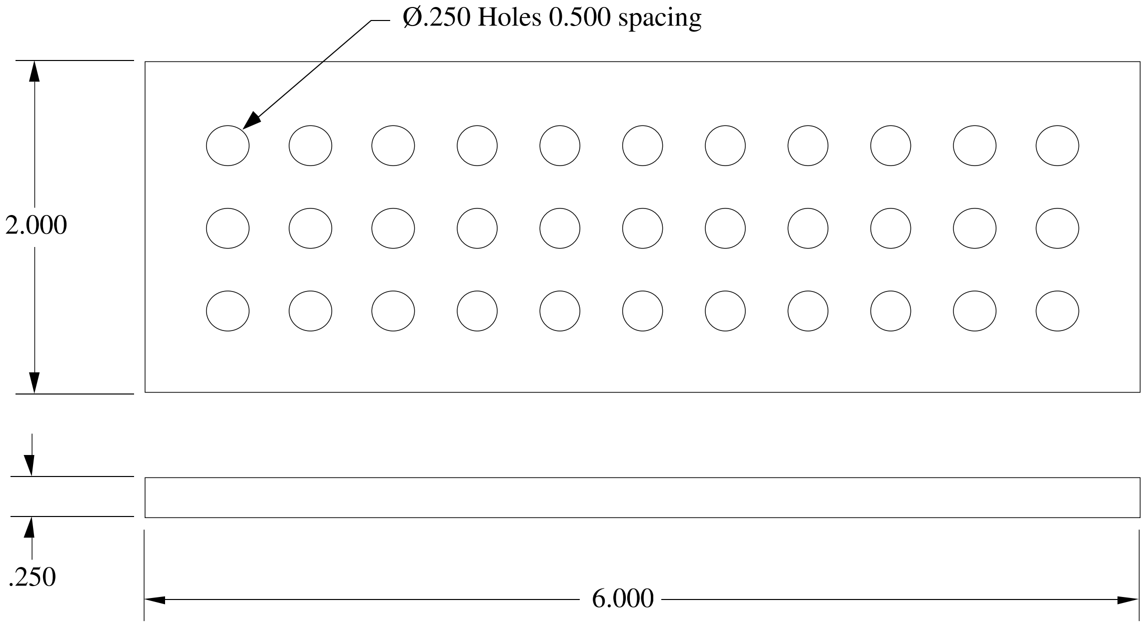


Figure 5-2: Aluminum Infiltration Base

Figure 5-3: Container for Embedding – Aluminum Base with Foil Walls

After the container is complete then the part must be embedded in wax and reference marks must be made. The following procedure is followed for embedding the part and is shown in Figure 5-4.

1. Fill the container with wax so that all the holes are filled and the bottom is just covered.
2. Before the wax completely solidifies, add the object and make a note of how it is placed in the bed.

3. Fill the container and generously cover the part. Make sure that the new wax is in the range 65-70 degrees C. This helps insure that the wax binds with the part. Keep adding wax as depressions form.
4. After several minutes the wax will be completely solidified. Drill two 1/4” holes for reference marks. Finish the hole to just above the base with a flat end mill. **Note**: Do not drill through the part.
5. Fill the holes using paraffin with a white dye added. Let the wax run into the hole so that it completely fills it and does not trap air bubbles.
6. Remove the foil and clean up any wax that may have leaked through foil or under the base.

After completing the steps, the part is ready to be scanned.

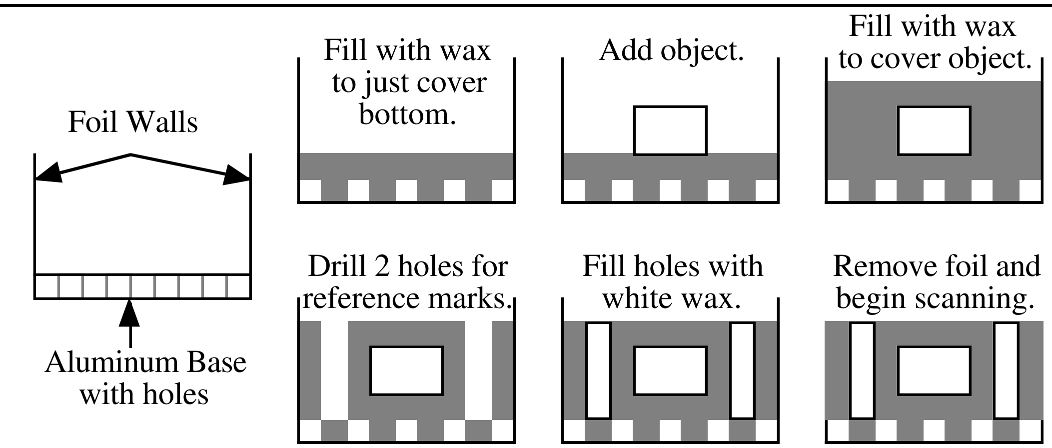


Figure 5-4: Part Embedding Procedure

5.3.2 Procedure

The scanning process is not currently automated, but it can be accomplished by hand using a milling machine with a fly cutter and a desktop scanner.

A thin layer is milled and the block of material is placed repeatedly on the same spot of the scanner using the lip of the scanner as a reference and is then imaged. This rough mechanical registration is necessary to aid in the digital registration and is explained in the Registration section on page 54. A milled layer of 0.01 inches is adequate for most parts. By initially assuring that the part begins parallel and is maintained parallel for each step, an accurate picture of each layer can be imaged. Nevertheless, errors in the same order of magnitude as the cutting thickness can occur unless great care is taken each time the part is refixtured in the mill. Therefore, cutting layers thinner than 0.01 inches is not currently recommended.

5.4 Thresholding

Each scanned image is a 256 gray level pixel map. It is important to decide what level of gray to threshold the image to distinguish the part and reference marks from the embedding material.

One method of thresholding is done using a pixel value histogram for all the images. The histogram yields two peaks, one light value for the part and reference marks and a dark value for the embedding material. A mid point between the peaks is used as the threshold value and each image is then converted to a bitmap.

Often the threshold value fell very near the midpoint of the gray level 128. Therefore, another equally valid approach to the threshold problem is to simply assume a midpoint at 128 and convert the images with that criteria. Computationally, it is significantly faster and the results are equally as good as the histogram method. Simply for time’s sake, the assumed midpoint method is recommended. However, in future applications where the problem may not be as simple as separating black and white, the histogram method would probably yield the best results.

5.5 Registration

One of the major benefits of the previously described digitizing method is that very fine details of the original object can be captured. While each layer can reveal very fine features, if they are not stitched together into a global picture then much of this scanning method’s desirable characteristics are lost. Furthermore, the part is destroyed with no possibility of regenerating it again. Registration is a technique implemented to maintain layer by layer information.

5.5.1 Mechanical

Mechanical registration involves creating an encasement for the part which has three orthogonal surfaces. The base of the block is used to keep the part parallel with the cutting plane. Thus the top and bottom are always parallel. There must also be two other perpendicular adjacent faces. Those two perpendicular sides are placed against the corner of the scanner creating a physical registration for the most recently exposed surface to be scanned. Ideally this method is sufficient to register each of the cuts to the other. Unfortunately, it is very difficult to create three perfectly orthogonal faces. The two opposite parallel faces are easily created by milling with a fly cutter. The two perpendicular sides however are only possible with extremely careful milling. Even with careful milling, the more typical result are sides which are almost perpendicular thus creating a gradual slant in the scanning as that side is used for registration. Also, placement on the scanner is subject to error each time. This may cause layers to be translated or rotated by several pixels from the previous scan. A rotation of just 1 degree in placement at the top of a 5 inch part can make the bottom off by almost 0.1 inches. This is a very unacceptable error.

Through filtering (see 6.2.3 Noise Filter p.66) errors on a local level can be hidden but not completely corrected. Thus regeneration of the part can be accomplished but with noticeable error. If there is a global error such as the one of the vertical planes not being true, then there will be an obvious bias in the regeneration.

5.5.2 Digital

With further development work, mechanical registration may eventually yield the necessary accuracy needed for this process. However, using a digital method in conjunction with the mechanical registration allows for fine tuning the alignment of each layer. Therefore, as long as the placements of subsequent scans do not significantly change spatially, meaning more than an 1/8”, then digital reconstruction is still possible.

5.5.2.1 Method

The digital method requires the placement of two registration marks perpendicular to the cutting plane. These marks are used to align the images. The challenges facing this type of alignment are fourfold. Each registration must be perfectly perpendicular to the cutting plane. The marks must be located. They must be uniform cross sections along the scanning volume and must be distinguishable from object being scanned.

5.5.2.2 Two or Three Colors

If the registration marks, embedding material, and part were each different and distinct colors or a different and distinct shade of gray then finding and separating them is an easy task. The problem lies in separating the different boundaries without any loss in border definition. If we lose the border definition then the digital alignment proves no better, in fact worse than the simple mechanical registration.

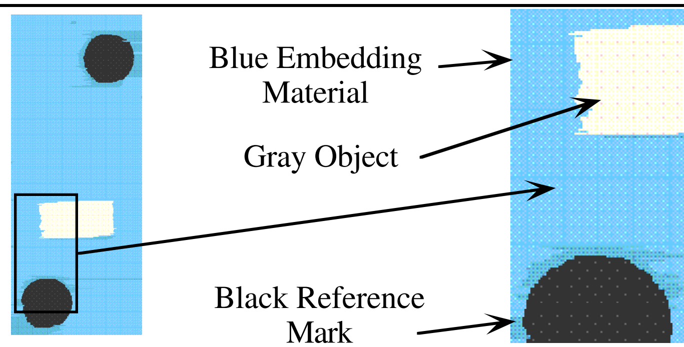


Figure 5-5: Three Color Reference Marks

A key element to the embedding material is a high contrast from the part. This allows for clear border recognition. This property of the embedding material must be maintained. Examine a sample from a 3 color scheme. Borders of both the part and registration marks are unclear.

Recognizing that the registration marks and the part never occupy the same position in a cross section, both the part and the registration marks can be the same color which is high contrast with the embedding material. This, however, presents a new problem, namely finding the registration marks.

5.5.2.3 Search Algorithm

Since the registration marks and the part are the same color, there needs to be a method for distinguishing between them. Only the first set of reference marks need to be found because all others marks are roughly in the same location because of the mechanical registration. Fortunately, there is much information known beforehand about the locations and size of the reference marks and many search techniques could be used to find them. One technique is implemented using a genetic algorithm (GA) optimization. A GA in this case is an overkill solution for the complexity of the problem. The GA performs adequately and finds the rough center of the marks in approximately one minute and that is all that is needed here.

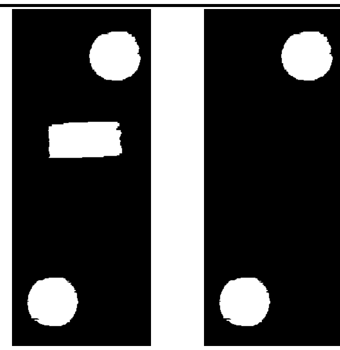


Figure 5-6: Finding Reference Marks

A very simple method that proved extremely efficient as well as effective must not be overlooked. Often a little user input can make automation much more effective. In the case of finding the registration marks, displaying the first image and asking the user to select a point inside the marks worked nicely. Finding the center of each mark only requires one seed value inside the mark. This time the human beat the machine hands down.

5.5.2.4 Finding Absolute Centers

As mentioned above, the genetic algorithm was originally implemented to find the exact center of the registration marks. However, to find the center of the marks, only one "seed" point is needed. From the seed pixel, a recursive algorithm finds the rest of the pixels in the registration marks by locating all connected white pixels. The center can now be found by a traditional center of area calculation as seen in Equation 5-1.

$$\bar{r} = \frac{\int \tilde{r} \cdot dA}{\int dA}$$

Equation 5-1: Center of Area

For digital images, the center of area must be calculated using a discrete form of Equation 5-1. This is seen in Equation 5-2.

$$\bar{r} = \frac{\sum \tilde{r} \cdot \Delta A}{\sum \Delta A}$$

Equation 5-2: Center of Area for Digital Shape

5.5.2.5 Alignment

Finding the centers of the reference marks is an important step. This is the beginning of aligning each of the images. With the four center coordinates alone, alignment is over constrained. This problem is similar to a classical dynamics problem of generalized coordinates for a rod in a plane. The solution for the rod problem is to have three generalized coordinates. Two of the coordinates come from one end of the rod and the other is the angle the rod makes with an arbitrarily defined fixed angle. As shown in Figure 5-7, the alignment for the images follows similarly with three generalized coordinates. Two coordinates are taken from the center of the first registration mark and the third coordinate is the angle from vertical that the two registration marks make with respect to each other.

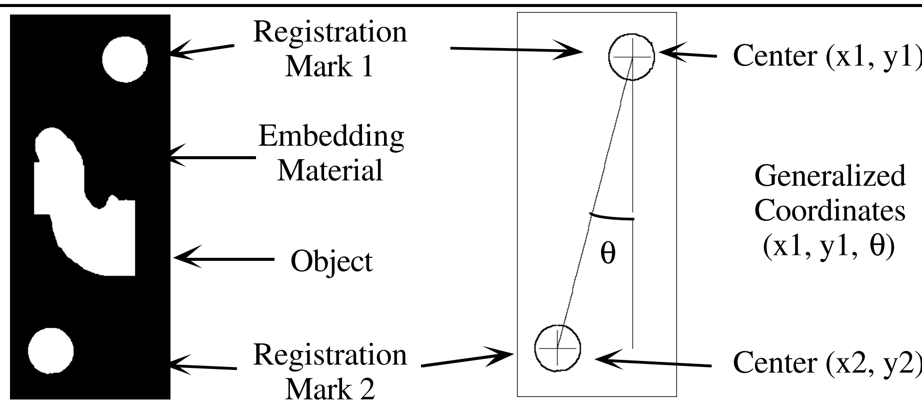


Figure 5-7: Generalized Coordinates for Orientating Images to Reference Marks

Each image is aligned using the three generalized coordinates (x1, y1, θ). The first task is to translate each image so that the first two coordinates match. Translation is much less interesting than rotation.

In computer graphics, rotation of objects and images is usually performed by applying the rotation matrix (Equation 5-3).

$$R(\theta) = \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix}$$

Equation 5-3: Rotation Matrix

This works perfectly well for single points, lines, and polygons but it is inadvisable to simply apply the rotation matrix to each pixel in a bitmap image. The problem lies in that there is not always a one-to-one correspondence between pixels in the source image and pixels in the rotated destination image. Because bitmaps are discrete, sometimes in rotation two pixels will be mapped to the same location in the destination image. The rotation will map white pixels to a black destination image and because there is not a one to one mapping, the destination image will have holes. Figure 5-8 demonstrates this problem. This happens even if the source image is completely filled in. Greater rotation angles correspond to even bigger problems with a maximum 45 degrees rotation. Higher angles will have less error but this pattern will cycle every 90 degrees of rotation.

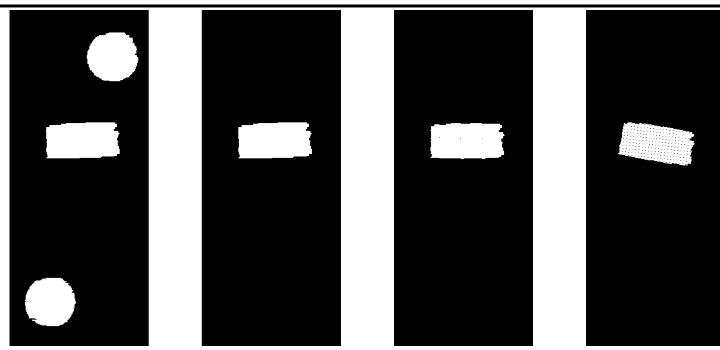


Figure 5-8: Rotation Problem – Holes in the Rotated Image

The solution to this problem is subtle. Instead of rotating and mapping each pixel into a new image, it is better to simply raster over the destination image and find which source pixel to copy by using the negative of the rotation angle. This will maintain solid white areas even without a one-to-one mapping. This is a simple point of view shift but solves the problem.

The process just described may seem like we are getting something for nothing. In actuality we are giving up one thing. If in the original image there were fine features that were only one pixel big, then they may be lost in the back rotation. However, the fine features that would be lost using this transformation are incapable of being printed by the 3D printing process or are probably noise. Therefore, we are sacrificing noise and impossible to create geometry in order to accomplish a much cleaner rotation.

To make this process even more computationally palatable, a quick raster rotation algorithm is used [Paeth]. The essence of the algorithm comes from the observation that the rotation matrix, R(θ) can be formed by multiplying three matrices which are shown in Equation 5-4 and Equation 5-5. These are the equations used for rotation in this research.

$$R(\theta) = M1(\theta)M2(\theta)M3(\theta)$$

Equation 5-4: Rotation Matrix Simplification

$$M1(\theta) = \begin{bmatrix} 1 & -\tan\left(\frac{\theta}{2}\right) \\ 0 & 1 \end{bmatrix};\ M2(\theta) = \begin{bmatrix} 1 & 0 \\ \sin(\theta) & 1 \end{bmatrix};\ M3(\theta) = \begin{bmatrix} 1 & -\tan\left(\frac{\theta}{2}\right) \\ 0 & 1 \end{bmatrix}$$

Equation 5-5: Simplified Rotation Matrix Elements

While aligning the images, the registration marks are removed or blacked out because they are not needed hereafter. Upon completion, the data now represents an accurate discrete digital version of the original physical model. This model is now ready for editing.

Chapter 6

6. Editing Electronic Model

There is a fuzzy line between the transition of reverse engineering and editing the electronic model. In this thesis, a CAD model is defined as an electronic model that can be altered then exported for manufacture (in this case 3D printing).

6.1 CAD Model Representation

After the realignment of the images is complete, the data now represents an accurate discrete digital version of the original physical model. Currently, this representation, called a voxel (combination of volume and pixel) model, is receiving much attention with most of that in the medical fields as a new visualization tool for radiologists. It has also been looked at as a CAD tool for mechanical parts. In the Rapid Prototyping field, the final file format to drive the rapid prototyping machines is, in essence, a compressed voxel model of raster information. A voxel model fits the criteria for a valid CAD model and therefore must be mentioned here.

One might conclude that in this design cycle there is no need to change existing information to any other file format besides a voxel. Voxel models can easily accommodate Boolean operations, are good at file-to-file comparisons, interference checking, offsetting, and eliminate many of the problems with surface models such as overlapping surfaces. Unfortunately, voxel models require an immense amount of memory (RAM) in order to work with the model interactively. This is difficult especially if the model has very fine resolution because the file size goes up by resolution cubed. The machines needed to run such models are still expensive beyond the scope of this design cycle's target. Therefore, while in the future voxel models may become the better CAD representation for this process, its major drawbacks far outweigh the proposed benefits. Compression schemes such as Octree [Zeid] or Wavelet [Strang] exist. But, in order to use voxels at their fullest potential, compression must not be done. Time will tell its outcome but right now a different type of model is necessary.

The CAD model representation that this research achieves is a commonly used file type called a boundary representation or Brep solid model. A Brep solid model is a closed surface model which means that each surface knows its neighbor and it is possible to traverse over the boundary of the entire object.

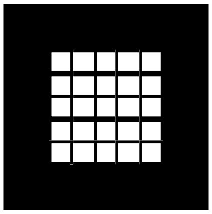
6.2 Boundary Points

6.2.1 Traditional Edge Detection

In section 5.4, the 8-bit gray scale images are simplified to a one bit representation. This helps the edge detection process dramatically. A traditional 2D edge detection algorithm uses a weighted sum filter like the one shown in Equation 6-1 where the a's are equal to one for on and zero for off. The final sum will always be an integer value. If the sum is one or greater then the pixel remains on and if the sum is zero or less then the pixel turns off. More elaborate schemes exist but this filter shows the simple essence of all similar algorithms. An example of this filter is seen in Figure 6-1.

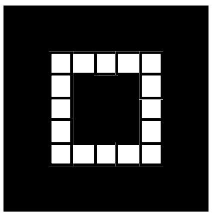
$$\begin{pmatrix} -a_1 & -a_2 & -a_3 \\ -a_4 & 8\cdot a_5 & -a_6 \\ -a_7 & -a_8 & -a_9 \end{pmatrix} \Rightarrow a_5 = 8\cdot a_5 - a_1 - a_2 - a_3 - a_4 - a_6 - a_7 - a_8 - a_9$$

Equation 6-1: Bitmap Edge Detection Filter



0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	1	1	1	1	1	0	0
0	0	1	1	1	1	1	0	0
0	0	1	1	1	1	1	0	0
0	0	1	1	1	1	1	0	0
0	0	1	1	1	1	1	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0

Apply Filter

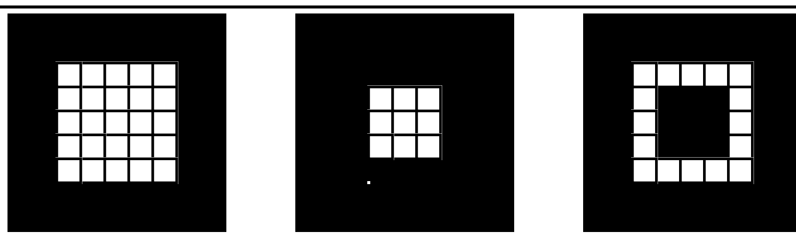


0	0	0	0	0	0	0	0	0
0	-1	-2	-3	-3	-3	-2	-1	0
0	-2	5	3	3	3	5	-2	0
0	-3	3	0	0	0	3	-3	0
0	-3	3	0	0	0	3	-3	0
0	-3	3	0	0	0	3	-3	0
0	-2	5	3	3	3	5	-2	0
0	-1	-2	-3	-3	-3	-2	-1	0
0	0	0	0	0	0	0	0	0

$Black \leftarrow if \leq 0 \qquad white \leftarrow if \geq 1$

Figure 6-1: Example of Traditional Bitmap Edge Detection

Edge detection of bitmaps can also be accomplished using a method called erosion. Erosion is accomplished by expanding each black pixel to its eight neighbors regardless of whether they are white or black and saving that new image in a separate file. Thereby "eroding" one pixel width or in other words the edge pixels of the white image. This eroded file is then subtracted from the original image which leaves the edge of the original image because the difference of the two images is the eroded pixels. This edge detection method is demonstrated in Figure 6-2. An adapted 3D version of this method is used in this research.



Original Bitmap Eroded Bitmap Subtraction Result (Edge)

Figure 6-2: Edge Detection Using Erosion

Regardless of which edge detection method is used, two fundamental problems exist. The first is that 2D edge detection looses information for surfaces which are scanned parallel to the cutting plane. The second is noise in the data after edge detection. Both of these problems make it difficult to construct a 3D surface model and are talked about later.

6.2.2 3D Edge Detection

The 2D edge detection schemes talked about in the previous section work on individual images separately. For parts which have low spatial frequency (curvature changes very slowly) such as some free form objects, this situation would be fine. However, this requirement is a severe limitation to part geometry. In the case of the example block shown in Figure 6-3, it has high spatial frequency at all sharp corners. Because edge detection is done on the images individually, they will not contain any point information for those features parallel to the cutting plane such as the top and bottom. Therefore, when surfaces are matched to the points, the top, bottom, and all features parallel to the cutting plane will be lost. Even though the block is a simple part, it shows the same problem that more complex parts have. Figure 6-4 illustrates this problem using the simple block shape.

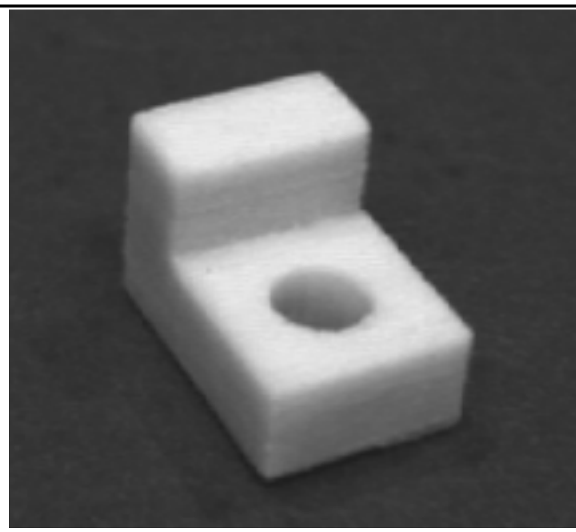


Figure 6-3: Simple Block with High Spatial Frequency

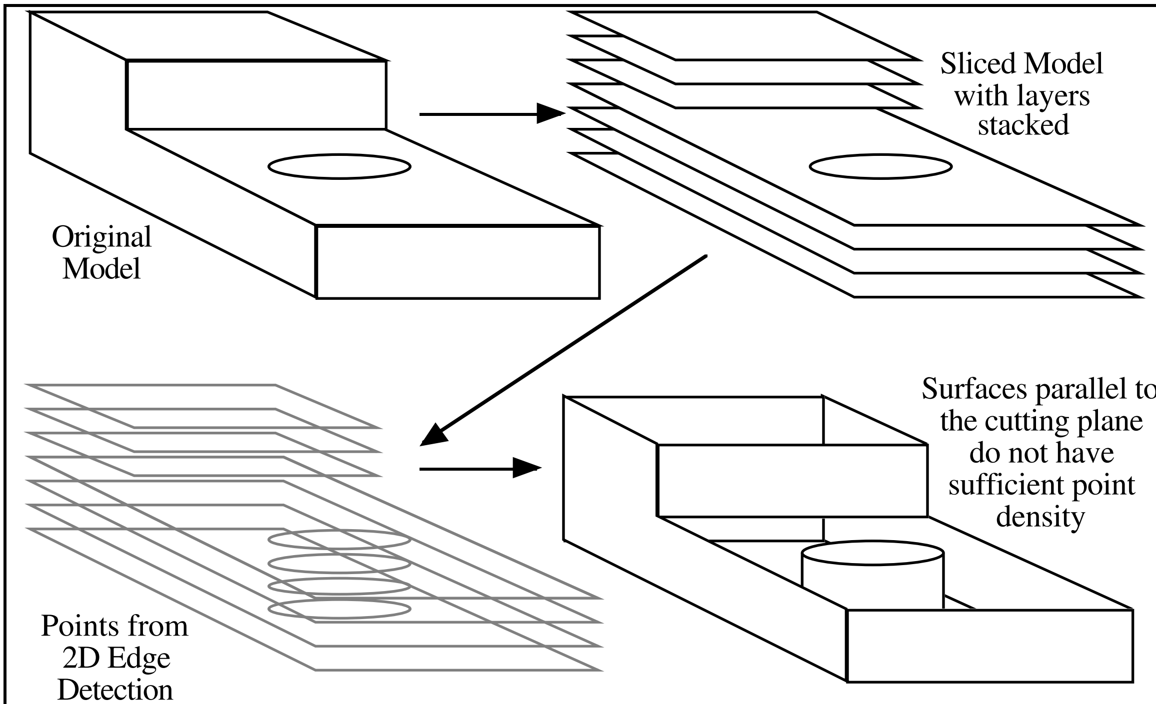


Figure 6-4: Surfaces Parallel to the Cutting Plane Lose Information with Simple 2D Edge Detection

The most straightforward way to solve this problem is to adapt one of the previously described 2D algorithms to three dimensions and deal with the discrete part data as a whole rather than slice by slice. This way the information about the surface points parallel to the cutting plane is maintained.

The first algorithm that uses the weighted sum can be altered to a three dimensional matrix of twenty seven elements. The center coefficient is 26 and all the rest are -1. Erosion can be done by having each black pixel expand into its eight pixel neighbors and the nine directly above and directly below. The method used in this thesis is a modification of erosion. Instead of expanding into all 26 adjacent pixels, expand the pixel into only the six pixels along the three major coordinate axis as is shown in Figure 6-5.

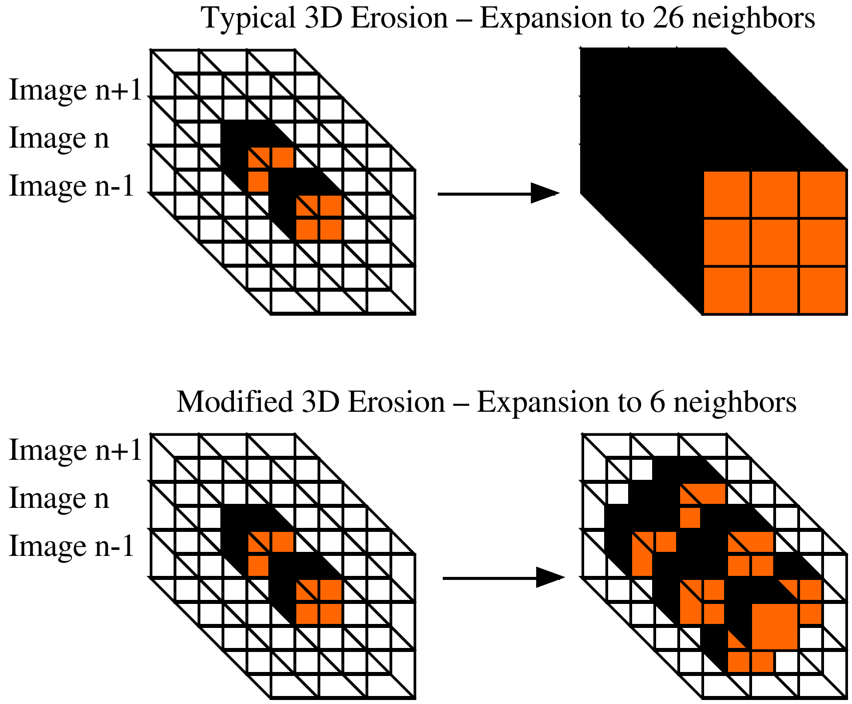


Figure 6-5: Modified Erosion

This modification does not adversely affect the outcome of the subtraction. In fact, it decreases the required CPU time.

6.2.3 Noise Filter

As with any edge detection algorithm, noise is always a problem. Little specks in the images will remain after edge detection is complete. As far as the filter is concerned, noise specs are edges. The surface model construction talked about in section 6.3 can accommodate small variation in accuracy of the data points on a surface but performs poorly with gross errors.

A noise filter that isolates solitary pixels and eliminates them is called a despeckle filter. It works in two ways. If a white pixel has no white neighbors then it is removed. Likewise, there may exist a black point within the boundary of the part and it can be removed with a similar technique. This despeckle filter is found in such image processing applications as Adobe Photoshop. This filter works fine if the noise in the images only comes from single pixel error. But, if the noise is on the order of two or more pixels then they are not removed using the despeckle filter. Since noise is usually always bigger than one pixel in the scanned images, the despeckle filter is not used in this research.

Interestingly, it is possible to make a better noise filter using the 3D erosion algorithm described in the previous section. Erosion, along with its opposite partner algorithm dilation, can remove substantially large noise speckles while maintaining part integrity. Dilation is just like erosion except that it expands each white pixel to its 6 closest neighbors. The following step by step algorithm which uses erode and dilate eliminates noise both black and white from the image information. It finishes with subtraction as described in the previous section to get the surface data points.

1. Erode – Eliminates white noise in black regions.
2. Dilate – Returns images to original state without the white noise.
3. Dilate – Eliminates black noise in white regions.
4. Erode – Returns images to original state without the black noise.
5. Erode – Removes white edge pixels.
6. Subtract – Produces surface data point information.

This idea is difficult to demonstrate in three dimensions therefore Figure 6-6 illustrates these steps in two dimensions. The erosion and dilation is done by expanding to its four neighbors. Erosion first removes the noise outside the part. With two dilations, the noise inside the part is also removed. The process is finished off and the boundary points are found with two more erosions and a subtraction. Refer back to Figure 6-5 to visualize how the 3D version would work.

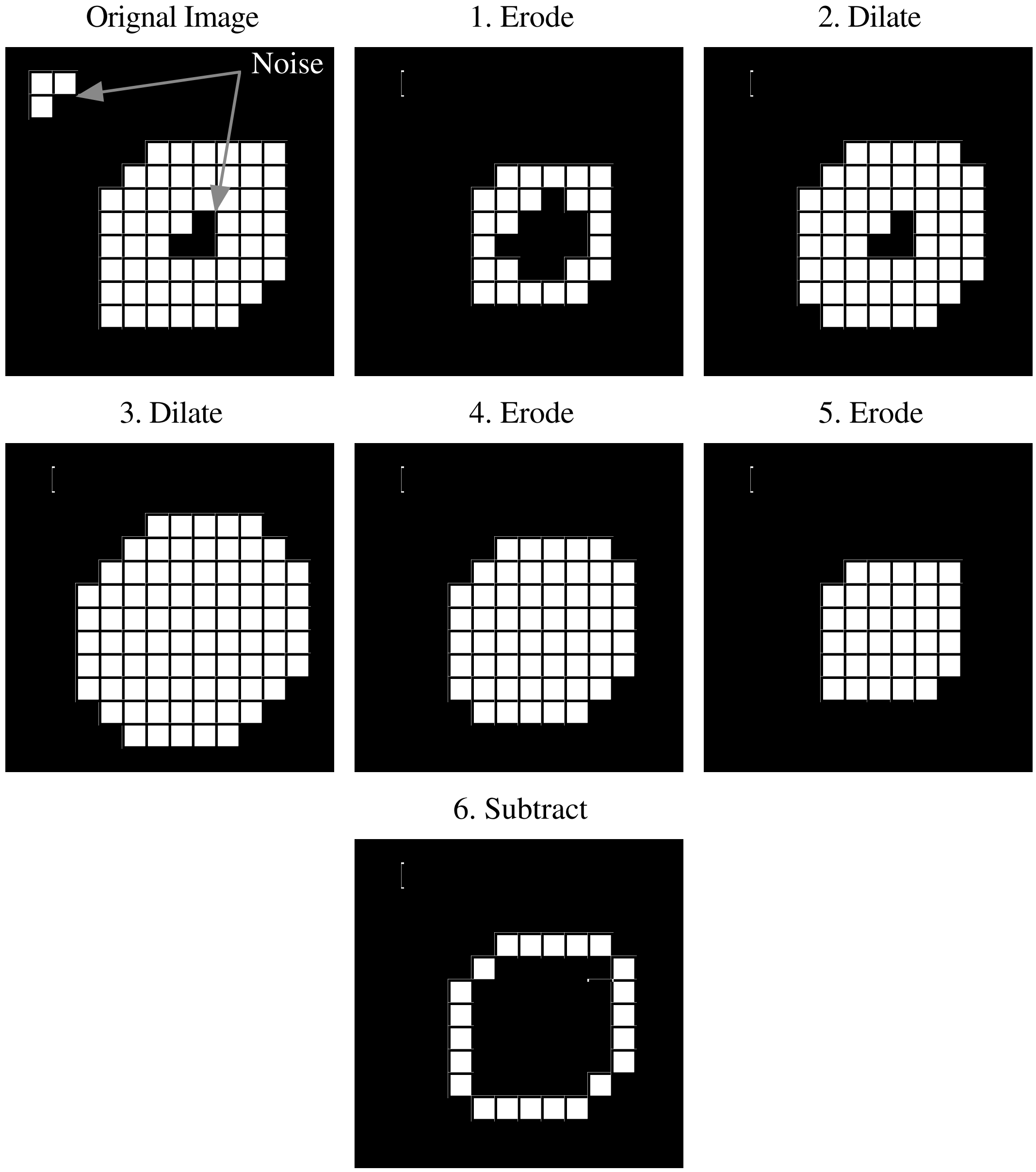
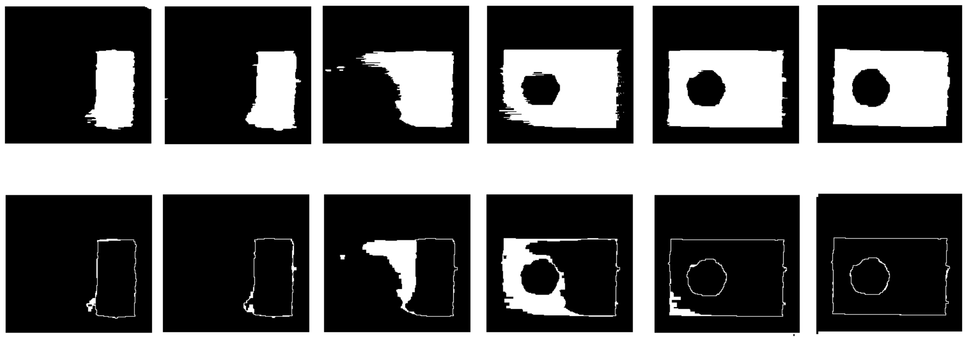


Figure 6-6: Noise Filter Using Erosion and Dilation

This example is only to demonstrate the concept of the filter. The real data sets are hundreds of pixels in all three dimensions. This example also clearly shows one remnant of the filter and that is the missing pixel on the two 90 degree corners. In the three dimensional filter, any pixel edge point at a sharp 90 angle will not be left after the filtering and edge detection. While this may seem like a problem, it is not because there is still ample information to create the geometry of the part.

Original Images



Boundary Images

Figure 6-7: Selected Original and Boundary Images from 3D Edge Detection

Filtering and edge detection is a computationally simple yet long process. With reading and writing buffer information and error checking, it may take one hour of CPU time to produce a proper data set from a moderate sized part. Since the file size of the part increases by N^3, time is an issue which must be met if extremely large parts are going to be designed with the design cycle. Selected images of a digitized block in original and boundary form are shown in Figure 6-7. Notice the middle images where the information at the cut in the block is retained.

6.2.4 Point Reduction

After the boundary points have been found, a surface model can be created. However, eliminating redundant point information is very helpful in simplifying the construction process. Take for example the boundary points found for the block (see Figure 6-8). For this particular block there are 54,716 points on the surface boundary. This is far beyond what is necessary for this simple block shape.

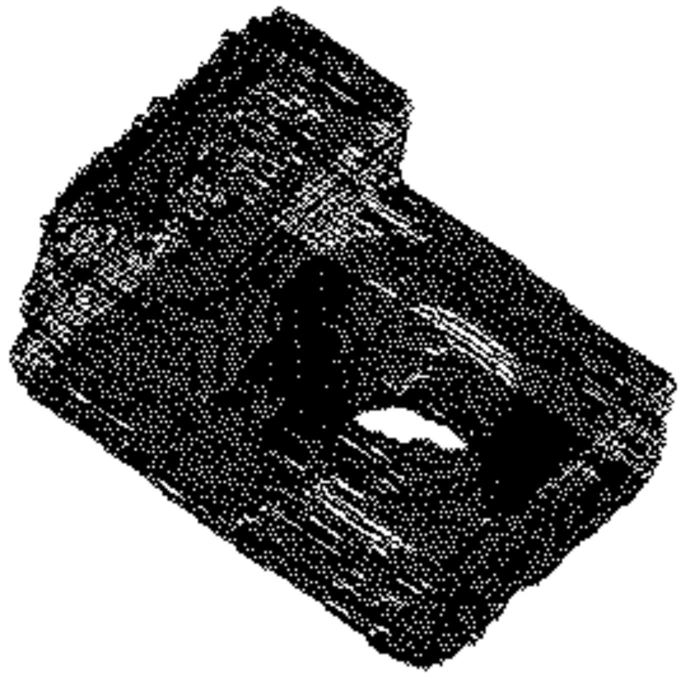


Figure 6-8: 54716 Boundary Points for the Digitized Block

In section 6.3, Points to Surface Model, there is a parameter which must be intelligently set. This parameter is the minimum distance between two parallel surfaces. Recalling the idea of minimum sample rate from signal processing, the Nyquist sampling frequency must be 2 times the minimum frequency component of a signal. It is also recommended that for good measure and to help eliminate magnitude attenuation, the minimum sampling frequency should be 4 times to 5 times the minimum frequency component of a signal. In this research the frequency is surface frequency and the period of minimum surface frequency is the minimum distance between two parallel surfaces. Therefore, there must be at least point density 2 times minimum sheet spacing and better to have 4 times the point density.

$$n \leq \frac{(R)\cdot(S)}{d}$$

n = Axis Reduction Factor (Integer Value)

R = Resolution (i.e. 300 dots per inch (dpi))

S = Minimum Surface Spacing (inch)

d = Points between required between S (d=4 recommended)

Equation 6-2: Reduction Factor

Equation 6-2 can be used to calculate a reduction factor "n". "n" is an integer value which corresponds to the number of white pixels needed in succession along a certain axis (nx, ny, nz). The reduction algorithm works on one row, column, or vertical column at a time. For example purposes, a row reduction is explained. The row is offset from the top of the image by an integral number of ny. It is tested point by point for a white pixel. When a white pixel is found, the test skips the next nx pixels in that row then continues testing. At the end of that row, the tests begin in a new row which is offset ny down from the row just finished until the image is complete. Next a new image which is nz from the completed image is tested. This test must be done for each image in the three major axis: X, Y, and Z. It is important to note that setting up a simple cubic grid and testing for a white pixel at each node in the grid will not give proper results.

Using the block points as an example, the X axis and Y axis are scanned at 300 dpi and the Z axis is cut at 0.010 inches or 100 dpi. Therefore, if the minimum spacing is 0.12 inches and using d = 4, the reduction factor equation results in n = 9 for the reduction factor in the X and Y axis and n = 3 in the Z. The images are resampled at these intervals which give a good reduction of points. For the block there is almost 2 orders of magnitude reduction to 1784 points.

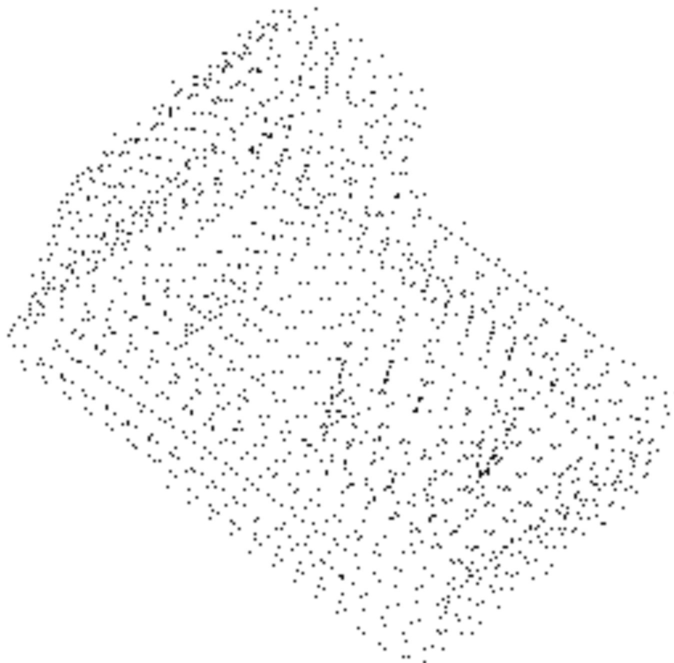


Figure 6-9: 1784 Reduced Boundary Points for the Digitized Block

6.3 Points to Surface Model

From 1991 – 1994 a group at the University of Washington worked on the problem of creating surface models from unorganized data points. Their method goes through three steps and each of these steps has been published in SIGGRAPH 92, 93, 94 [Hoppe et. al.] respectively. The first step is rough model generation which simply gets the topology of the data set, the second is an optimization of the surface by pulling the polygons found in step one closer to the actual data points, and the third is a piecewise smooth surface fit. The descriptions of the procedures is taken from the abstracts of the respective papers.

Step one, "takes as input an unorganized set of points x_1,...,x_n subset IR^3 on or near an unknown manifold U, and produces as output a triangulated polyhedron that approximates U. Neither the topology, the presence of boundaries, nor the geometry of U are assumed to be known in advance --- all are inferred automatically from the data." In other words the goal of this step is simply to determine the topological type of the surface and to produce a rough estimate of its geometry. The method for step 1 defines a function $f: R^3 \rightarrow R$ that estimates the signed geometric distance to the unknown manifold U, and then using a contouring algorithm (modified version of the marching cubes algorithm

[Lorensen]), extracts an approximation to its zero set, $Z(f)=\{p \in R^3 : f(p)=0\}$. The one user input to this step is the minimum surface spacing also talked about in Point Reduction section. Figure 6-10 shows the result of step 1 using the reduced data set taken from the block in the previous section.

Figure 6-10: Reconstruction (Step 1) – Generates Topology

With step 1 complete, step 2 attempts to solve the following problem: "given a set of data points scattered in three dimensions and an initial triangular mesh M-{o}, produce a mesh M, of the same topological type as M-{o}, that fits the data well and has a small number of vertices." Their objective function is defined as the minimum "energy function that explicitly models the competing desires of conciseness of representation and fidelity to the data." The variables for the optimization consist of the number of vertices in the mesh, their connectivity, and their positions. The optimization search space is all meshes of the same topological type as that produced in step 1 and this search space is traversed "using a set of mesh transformations (edge collapse, edge split, and edge swap) that locally modify the mesh connectivity." The optimization also tries to reduce the number of necessary polygons. As can be seen in Figure 6-11, this produces a much better model of what the data represents. Similar research was done at the Iowa State University. Their goal was also to simplify the polygon mesh while maintaining topological and surface constraints [Renze].

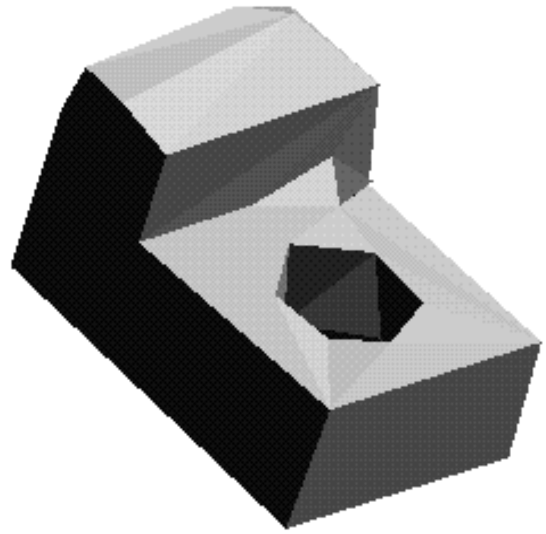


Figure 6-11: Optimization (Step 2)

The output of step 2 in many cases gives an adequate representation of the original model. However, for freeform parts, the large polygonal output may be somewhat “chunky” and a better representation is required. Creating a better representation can be accomplished with a possibility of two methods. Both are based on a new class of smooth surface representations. The first is a deformable surface patch [Celniker and Gossard] and the other is a surface representation based on subdivision [Hoppe et. al.].

The deformable surface patch uses a finite-element solution technique which yields a continuous mathematical description of a surface over a triangular region. This patch can be deformed using mathematical forces (pressure or point) and best fit to data points. This representation of the electronic model may in the future be implemented. However, current computer models in use with rapid prototyping machines are based on a polygonal representation. Therefore, when generating an electronic model from scanned data it is presently adequate to leave that model in polygonal form and the surface representation based on subdivision is used. This representation subdivides each polygon in the optimized mesh into several smaller polygons. This was first introduced at the University of Utah [Loop] and generalized to include rules for sharp corners by the group at the University of Washington [Hoppe]. The subdivided polygons can deform to better fit the data As can be seen in Figure 6-12, the subdivision creates a surface which closely represents the physical model. The visible error represents a bubble trapped on the surface of the part during embedding. (This scan was not done with the recommended materials discussed in chapter 5.) This error is corrected in Figure 6-15 on page 77.

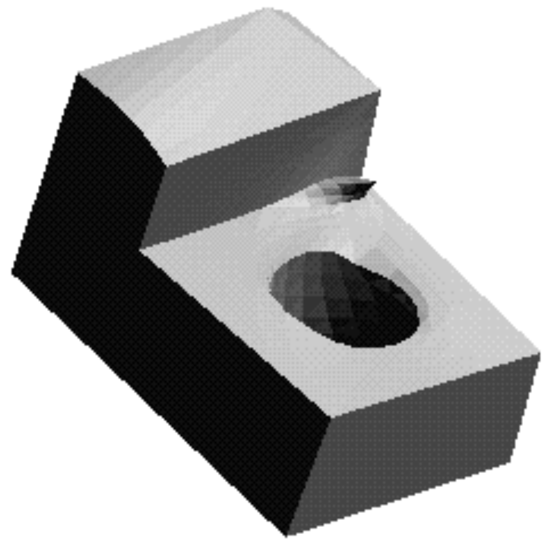


Figure 6-12: Surface Fitting (Step 3)

6.4 Surface Model to Solid Model

The output of the points to surface model section is a set of polygons which describe the model. If the three surface reconstruction algorithms start with a complete data set fully dense on all surfaces, the output describes a closed surface. Therefore the topology can simply be rescued from the polygons and put into a “useful” data structure. There is commercial software which works strictly with polygons. It is called FormZ created by Autodessys, Inc. based in Columbus, Ohio. This software puts the data into its proprietary format and then can be manipulated by a user. Since the data is in a proprietary format, further extension and research using FormZ can not presently be done and therefore is not used in this research.

In this work, the set of polygons is decomposed into points, edges, and polygons and written into a file format called TWIN developed by the CADlab at the University of Purdue. A sample of the TWIN file format is found in the appendix. The TWIN file format is a Brep solid model and the data is no longer a simple surface model. Once converted, the model is ready to be manipulated with traditional CAD tools such as Boolean operations.

6.5 Boolean Operations

Custom software with a user friendly interface is written in order to perform simple Boolean operations and other functions. The program organization and function capabilities are listed in Figure 6-13 and the user interface is shown in Figure 6-14. The program is subdivided into four menu options with several functions in each. Most functions are self-evident. The Information menu gives the status of various parts of the program an is useful for debugging purposes. The three dots after the functions note that there will be a pop up window or that the function will need user input to continue.

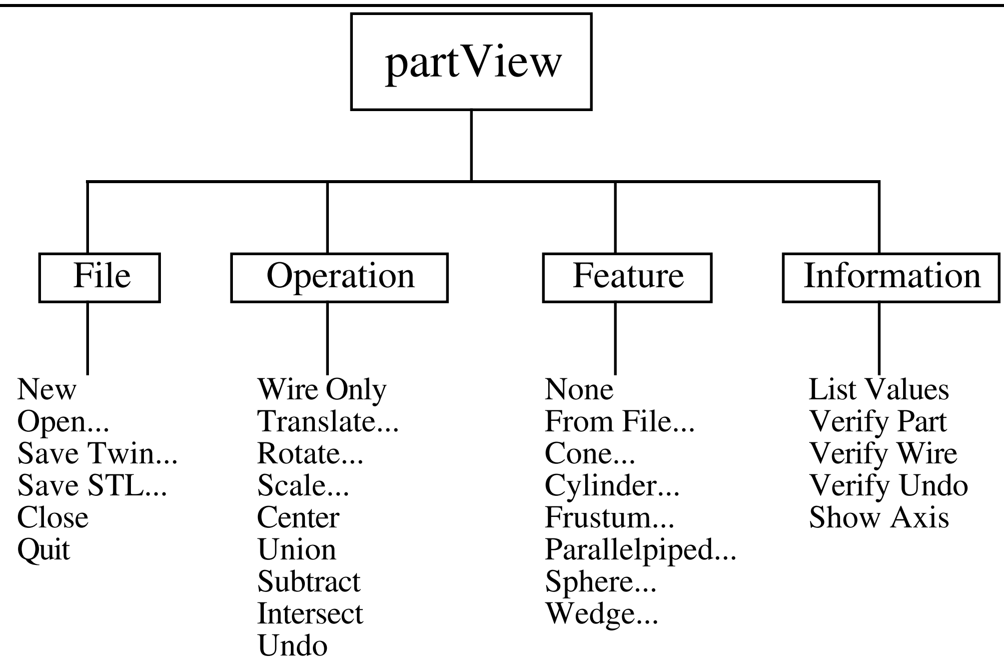


Figure 6-13: partView Program Structure and List of Functions

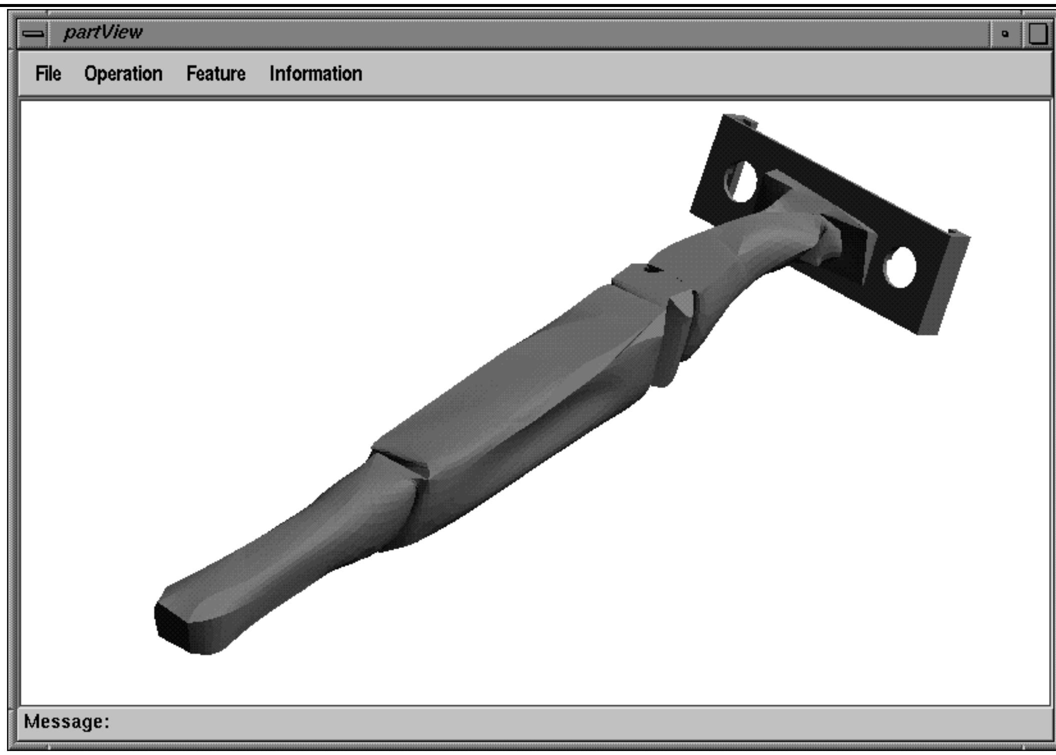


Figure 6-14: User Interface for CAD Operations (partView)

The partView software is capable of removing an error in the block as seen in Figure 6-15. This fix is done by subtracting a parallelpiped which is exactly parallel to the two surfaces that the error touches.

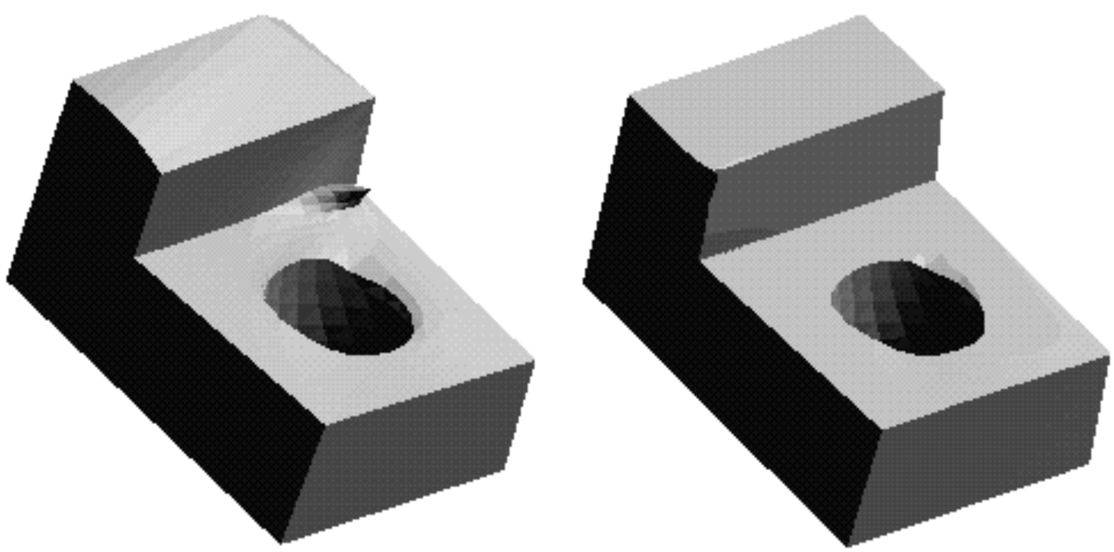


Figure 6-15: CAD Model Touch Up (Error Removal)

A better example of how the Boolean operations can be utilized is demonstrated in the next series of figures. Figure 6-16 shows a block and parallelpiped together and then a groove is cut by subtracting the parallelpiped.

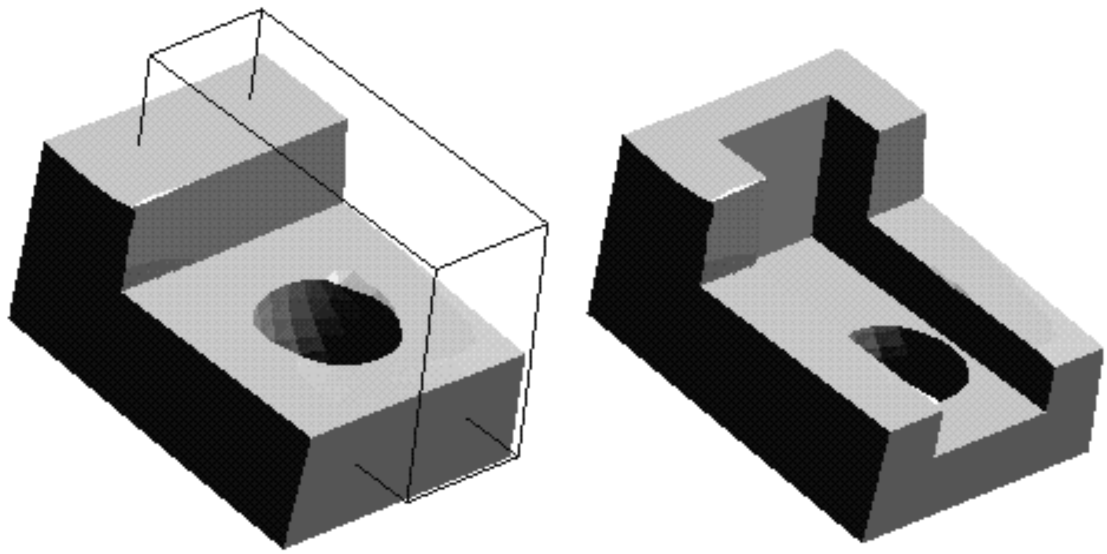


Figure 6-16: Set Subtraction of Parallelpiped

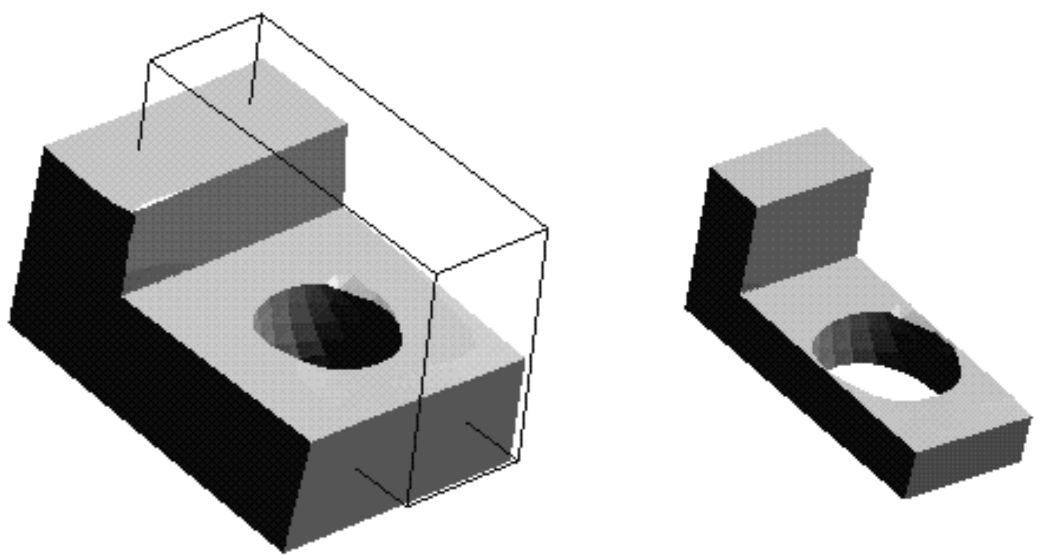


Figure 6-17: Intersection with Parallelpiped

By performing an intersection with the same setup, a slider which exactly fits in the groove can be created as seen in Figure 6-17.

Small knobs can be added to the back of the groove to help secure the slider in Figure 6-18. This is done by performing a union operation.

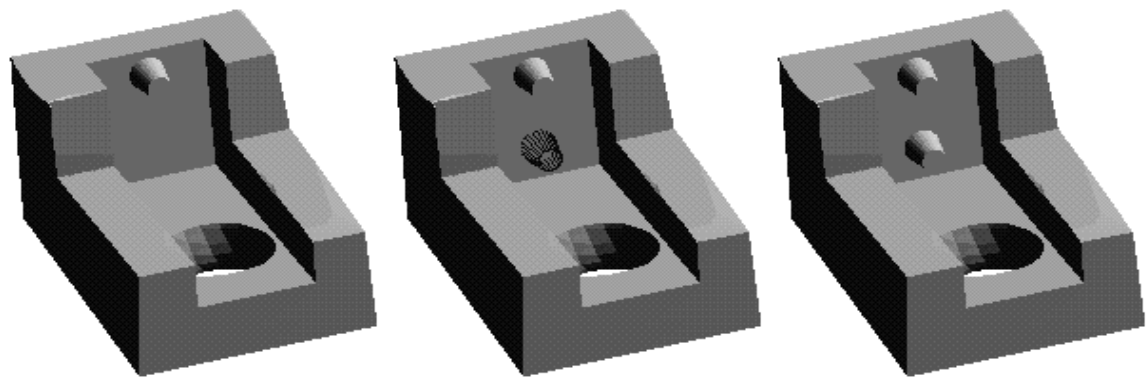


Figure 6-18: Union with Two Cones

Even though this is a simple example, it is important to note that one complete loop of the design cycle is now completed. The block started as a CAD model, physically created by 3D printing, digitized and returned to an electronic medium. Since these CAD models can be easily saved into the standard rapid prototyping file format .STL, the cycle can go on indefinitely.

Chapter 7

7. Applications

7.1 Subjective Design

One of the main applications of this design cycle is the ability to design parts which have both subjective contours and functional constraints. The design cycle has the unique ability easily work with basic geometric shapes as well as free form surfaces. The basic geometric shapes with precise mathematical descriptions like cones, cubes, etc. are best created in an electronic environment. It is difficult to create freeform shapes in the computer and is much more suited to creation by sculpting a physical model.

Many consumer handheld objects fit into this category of having subjective contours as well as geometric and kinematic constraints. Figure 7-1 and Figure 7-2 show three possible candidates: a handheld razor, a car door handle, and a screwdriver. Others examples include an electric razor, a computer mouse, and a telephone handset. There are many more.

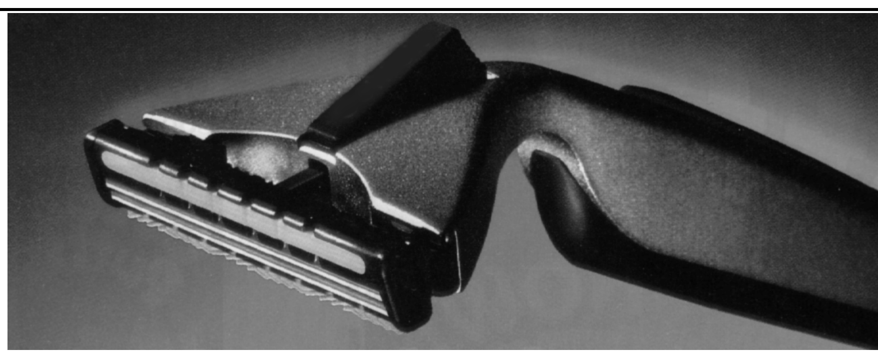


Figure 7-1: Handheld Razor

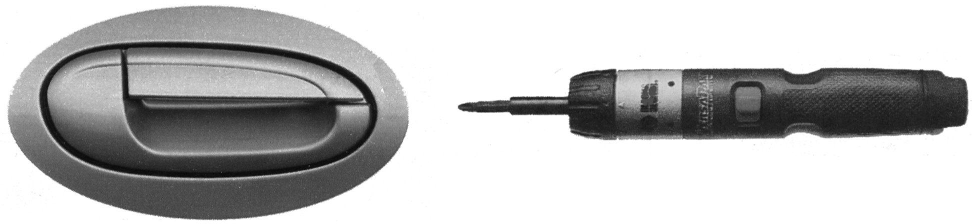


Figure 7-2: Car Door Handle and Screwdriver

The car door handle is a perfect example of aesthetics combined with functionality. The exterior must be both aesthetically pleasing as well as have the right “feel”. It is also connected to a kinematic linkage that opens the door. The look and feel of the door are subjective qualities and are best designed with a physical model. The linkage is made up of typical geometric shapes and is best created in the computer where dimensions can be defined precisely.

The design cycle also applies well to the creation of the screwdriver. While the screwdriver in Figure 7-2 is somewhat simple, the exterior could be very complex to make it more ergonomically sound. A parting line must be defined and mounts need to be created in the interior for the motor, battery, switches, etc.

As a demonstration of the functionality for the design cycle, it is used to create a new razor handle. The important part of the demonstration is not the finished razor. The important part is the design process and recognizing the contribution the design cycle makes to the design of complex 3D shapes.

The demonstration begins in the computer where the rough shape of the razor is quickly blocked out into a CAD model which took approximately 10 minutes. Using Three Dimensional Printing, the model is created in the glass powder and wax composition and an industrial designer carves and reshapes the model into a complex shape. When modification is complete, the physical model is digitized and converted into a polygonal Brep CAD model where the head of the razor is added.

Figure 7-3 shows the razor in its blocked out form as a computer model. Only major features are included.

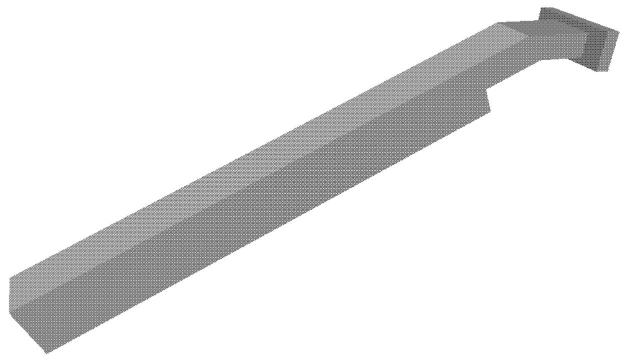


Figure 7-3: Beginning CAD Model

The CAD model is used to produce several physical models by 3D printing and wax infiltration (Figure 7-4). This gives the designer freedom to think and design freely.

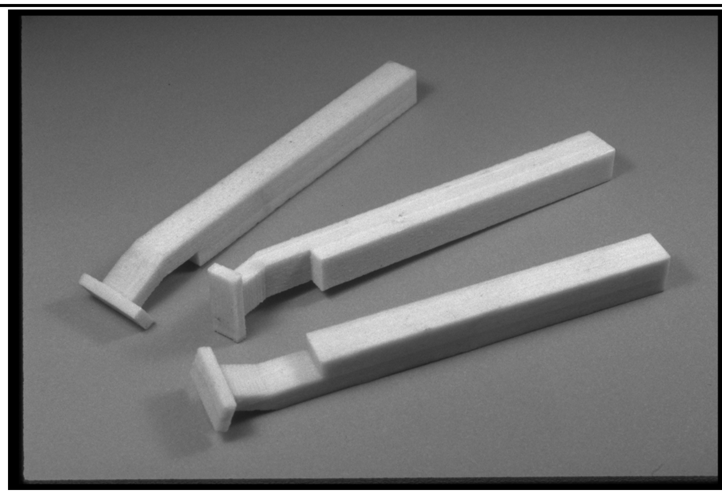


Figure 7-4: 3D Printed and Infiltrated Razor Handles

An industrial designer drew rough sketches of possible concepts then carved those ideas in Figure 7-5 and Figure 7-6. He used a simple carving knife and each razor took just under 1 hour to make.

Figure 7-5: Razor Modified Using Hand Tools

Figure 7-6: Second Modified Razor Using Hand Tools

The design shown in Figure 7-5 is picked as a good concept and is digitized. The slice thickness is 0.01 inches and each layer is scanned at 300 dpi. Just over half an inch (52 images) is sliced and scanned. Custom software called “boundary” is used on the images to align to the reference marks, filter out the noise, create a full boundary point representation and reduce the number of points required as defined by the minimum surface spacing. The output is 5942 points on the surface of the digitized part. Figure 7-7 show the data points after being reduced.

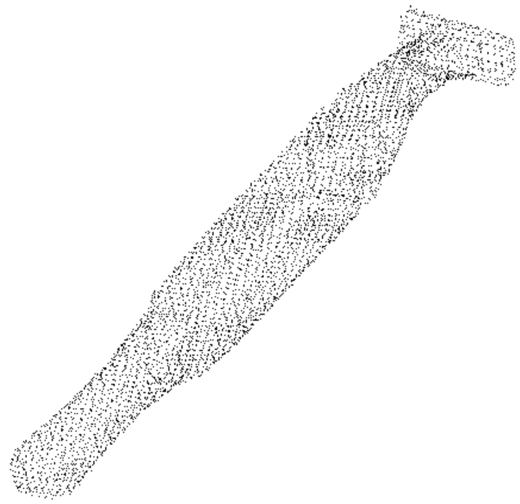


Figure 7-7: Digitized Data Points of Razor

A polygonal Brep CAD model is created from the data points by using the three steps of the reconstruction method discussed in chapter 6 and is shown in Figure 7-8.

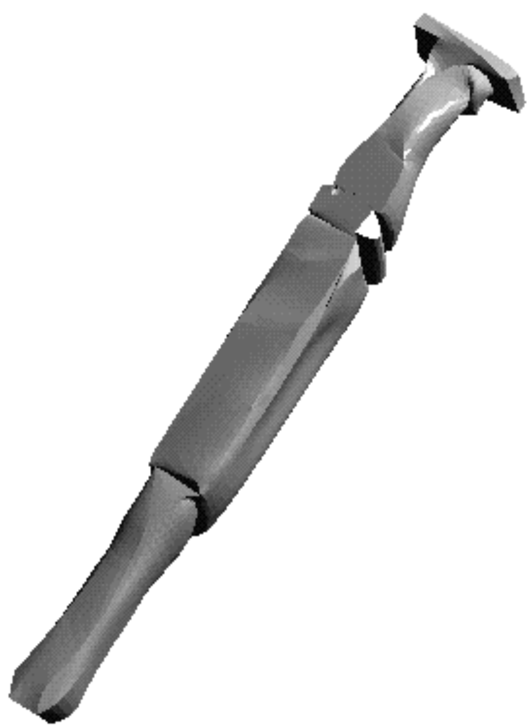


Figure 7-8: Razor CAD Model

The razor head in Figure 7-9 is a holder for certain types of disposable razor cartridges. For example, this head might be standard across all the razors for a certain. In order to complete one iteration of a design, the head must be unioned with the razor body design. The result of the union operation is shown in Figure 7-10.

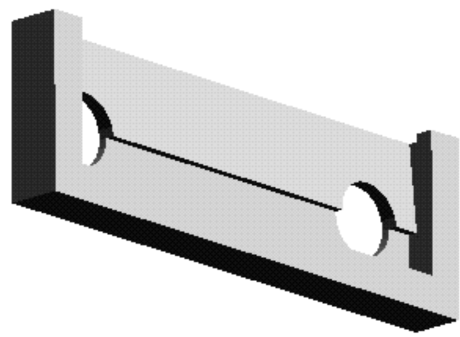


Figure 7-9: Razor Head Design

Figure 7-10: Razor Body Unioned with Head

This example demonstrates the use of the design cycle. It allows the designer to use the model, physical or electronic, which can best implement a design idea. The complex surface of the body is made with the physical model and the head is completely designed in the computer and both are combined into one model. This design is created quickly and efficiently. This gives the designer more time to design rather than spending time on figuring out how to design.

7.2 Assembly Design

As described in the previous section, subjective designs with constraints can be extended to assembly design as well. There are many assemblies that have two or more parts that mate and form a continuous subjective contour. Commonplace examples of this type of design are cameras, walkmans, cellular flip phones, and camcorders.

Figure 7-11: Subjective Shape Assembly Examples of the Design Cycle

For this type of design, the freeform shape can be designed irrespective of the assembly. Obviously, there are internal geometry constraints but as long as all internal features can fit inside a certain volumetric envelope then the exterior design can be placed in whatever manner on top of that envelope (see also section 8.1.3 for further discussion about internal constraints). After the exterior is designed, then the assembly features can be extracted.

This type of design can be applied to the camera shown in Figure 7-11. After the smooth exterior shell is created using a physical model, the shape of the flash and its pin connections can be created in the CAD model. This way, the assembly is still guaranteed to be flush. The same technique can also be applied to the film door in the back.

A walkman, diskman, flip phone, and camcorder can all be designed in a similar fashion. The most important point to remember is that the time necessary to do this design is shorter than the standard practice. Many designs can be explored quickly and efficiently.

7.3 Medical Visualization and Design

Physicians at the Massachusetts General Hospital and also at Brigham and Women's Hospital expressed interest in the design cycle. Medical visualization using high end computer graphics is becoming an important tool for doctors. Computers allow them to see and diagnose problems that were previously impossible. However, just like engineers, doctors sometimes need physical models to help them understand and visualize a problem and subsequently develop a solution. Physicians also use models to explain problems to their patients. For example orthopedic surgeons often have models of typical knees, elbows, and other joints to show patients what has happened to their body and then describe the surgical procedure that will be implemented. Internal medicine physicians have models of typical organs for the same purpose.

The design cycle can be used in the medical field for two purposes. One is visualization. Computed tomography (CT) scanning and magnetic resonance imaging (MRI) capture information about the human body in volumetric form. These two procedures capture information about the human body in volumetric from in a section by section method but is non destructive like the scanning method employed in this thesis. CT scanning, MRI, and destructive scanning all produce the same type of data set. Therefore, using the methods of the design cycle described in chapters 5 and 6, computer models of organs, bones, etc. inside a human body can be generated. Subsequently, these models can also be translated into physical models quickly and easily. Figure 7-12 shows a physical model of a human liver section generated from CT data. (The CT data is courtesy of Dr. Ron Kikinas and Dr. Michael Halle of Brigham and Women's Hospital.)

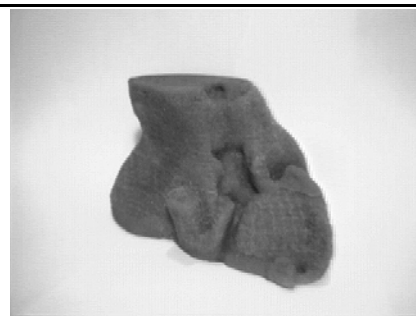


Figure 7-12: Physical Model of a Liver Section

Another medical area the design cycle can assist is in orthopedic and reconstructive (plastic) surgery. The design cycle makes it possible to produce custom implants for joint replacements such as knees or hips and also for plastic surgery implants. For example, a person might be involved in an accident and shatter the Zygomatic bone (cheek bone) on the left side. A custom implant can be made by creating a CAD model of the cheek bone on the right side and doing a mirror algorithm and subsequently creating the physical part. Even though human's faces are not symmetric, this model copied from the right cheek bone would be a much better starting point for the surgeon than nothing at all.

Whether or not there is a market for this type of custom work is another question especially with hospitals trying to cutback on medical expenses. However, 3D printing has unique manufacturing capabilities such as the ability to manufacture macro-texture on free form surfaces. Figure 7-13 shows two such examples. One is a torus with a mushroom field and the other is an artificial hip stem with texture to aid in bone ingrowth.

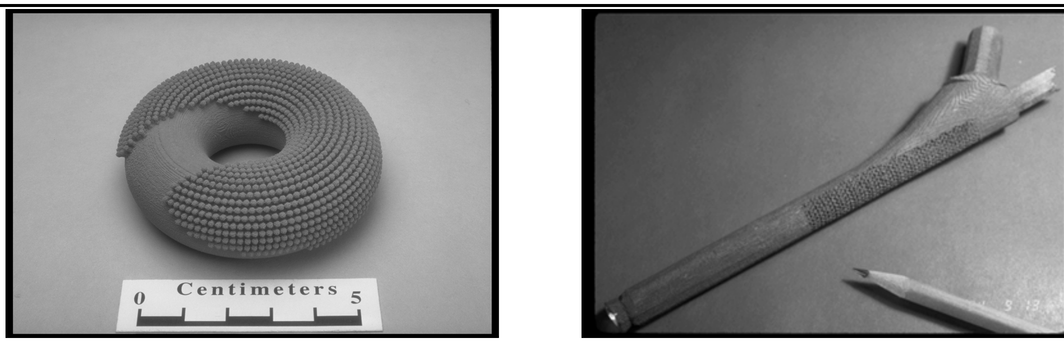


Figure 7-13: Torus with Mushroom Field and Artificial Hip Stem with Macro Texture

Further applications to the medical field are described in the future work chapter on page 97.

7.4 Design Iterations

The three previous examples of the design cycle have demonstrated short examples which may traverse the loop once. Design is not just a do it once and stop activity. It is naturally an iterative process. Design is done over and over. Changes are made to a design. A design may revert back to an older idea. A design may even be scrapped and a new one will start over fresh. The power of the design cycle allow a designer to enter the cycle with either of the modeling mediums (physical or electronic). Below are two more ways that the design cycle might be entered.

Design of a mating part – A designer may need to design a part to mate with another part that exists as either a physical or electronic model. An electronic model for a mating part can be used to establish boundaries on the part to be designed and the cycle can proceed. A physical model for the mating part can either be digitized or used to define the boundary of the physical part.

Redesign of a part – The designer starts by calling up the CAD model for the part in last year's model. Minor changes can be done directly in the electronic model and a physical mode can be printed for verification. Major changes might better be accomplished by printing a physical model of the part and then manipulating the physical model. The changes would then be digitized and captured in electronic form. Finally, precise specifications could be added electronically.

The second example is interesting because it gives various options for editing the design. That is the way the design cycle should be used. The design cycle need not always be followed in every case. Take for example the part shown in Figure 7-14.

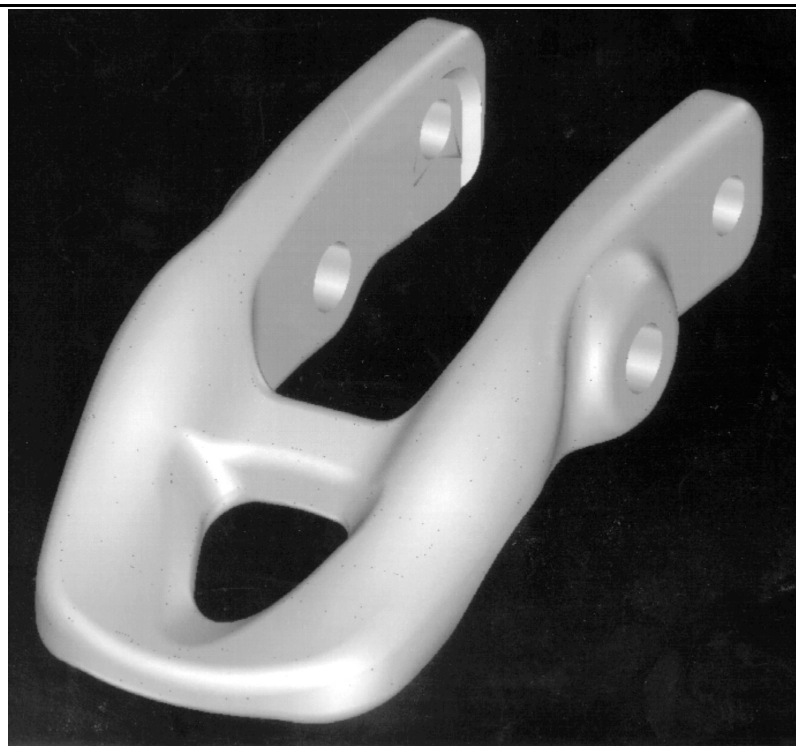


Figure 7-14: Complex Part from Ski Binding

The part may be printed out and the designer, in trying to place it in the proper assembly position, notices that one hole is the wrong dimension and in the wrong place. Rather than try to modify the physical model, the electronic models should be altered. This little jump bypasses one step in the design loop. Changes need to be made to the model that can most easily represent the changes.

The design cycle is not a tool which constrains the designer to work in one particular way. It is a powerful, liberating tool which allows the designer to be creative and to easily use the models at hand to express their concepts.

Chapter 8

8. Future Work and Conclusions

8.1 Future Work

Even though a working design cycle now exists and different applications can be explored, it is still in a fundamental state and many improvements and additions should be made to both software and hardware. Along with specific tasks that must be completed, ideas for further application can also be explored.

8.1.1 CAD

8.1.1.1 Shape Features

A major task will be to develop software methods to enable the user to identify and modify "shape features", e.g. planes, cylinders, pockets, etc. Such "shape features" are higher-level descriptors of the CAD model. They play an important role in efficient shape editing during CAD model creation and modification because they describe a model in a higher level language which is closer to the way that the designer thinks about it. Shape features also play an important role in down-stream manufacturing applications that use the CAD model.

Software tools can be envisioned which enable the user to designate distinct regions of the model, apply constraints to that region (e.g. planar, cylindrical, etc.) and have the model adjust to those constraints. Similar work is being done by researchers at the University of Utah.

For the design cycle extracting shape features from the reconstructed CAD model poses a particularly difficult task. Since a designer is modifying the physical model by hand, the intent of the designer may be different than the physical model created for prismatic shapes. For example if the designer intends to make two surfaces perpendicular, but actually carves them to be acute or obtuse, the reconstructed CAD model will have off-perpendicular surfaces. In this case the designer could edit the surfaces to be perpendicular, but creating CAD models which capture the designers intent from digitized data is an interesting avenue for future research.

Often physical and electronic models have a plane of symmetry. If such a model is carved by a designer, it may not actually be symmetric because humans have difficulty making perfectly symmetric objects. However, it is conceivable that the designer could define a plane of symmetry in the reconstructed computer model. The two halves of the object could then be adjusted to best fit the data points while maintaining the symmetry requirement. This is an example of capturing design intent rather than simply recreating a model from raw data.

8.1.1.2 Cycle Remembering

Finding shape features as described in the previous section might be semi-automated if the reconstructed CAD model had a method of comparing itself with the original CAD model that was used to create the physical prototype. In other words, only the changes made to the physical model would be registered and modified and the rest of the model would revert back to the original CAD model. This would hopefully create a more simple CAD model as well as reduce time needed to extract shape features.

There are two problems that must be confronted in order to remember cycle iterations. The first is how to align the new model to the old. Depending on how much the physical model is modified this problem could be as simple as having the user define three common points if there are not significant changes. If the digitized model is significantly different from the original one then the solution may not be so easy. Regardless of the solution to this problem it will need user interaction and will be difficult to automate.

The other problem is the comparison to identify which parts have been modified. A possible solution to this is to create offsetting surfaces within a tolerance of the original model and build a “similarity envelope” where the two models can be compared and differences can be identified. This idea is shown as a two dimensional example in Figure 8-1.

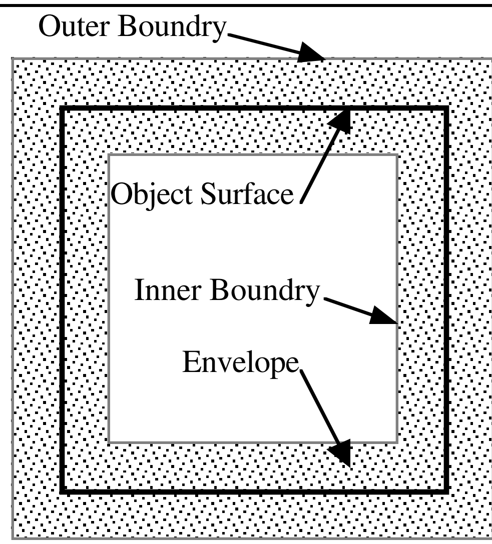


Figure 8-1: Similarity Envelope to Recognize Altered Physical Model

Unfortunately, offsetting surfaces of an object is well known as a difficult problem. There are several problems. Two of which, overlapping surfaces and touching surfaces, are shown in Figure 8-2. These are similar problems that the rapid prototyping standard CAD file, .STL, has as well. The first is overlapping surfaces and the second defines two surfaces in the same location with normals in opposite directions.

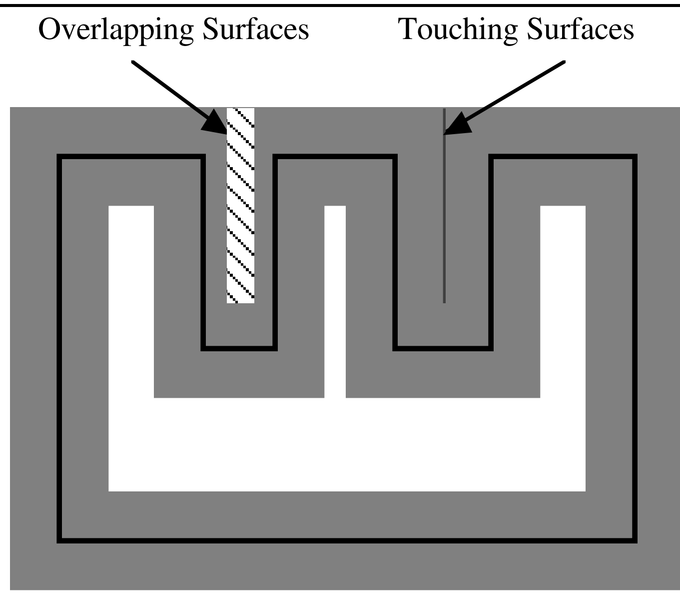


Figure 8-2: Offsetting Surfaces Problems for Similarity Envelope

Inventing solutions to this problem is a complete research topic in itself. However, one such solution is to take advantage of the fact that the raw data of the digitized object comes as a discrete (voxel) representation. In order to build a "similarity envelope", the original CAD model could be discretized itself. Then, using the erosion and dilation techniques described in chapter 6 on page 64, a big model and a small model could be created and the subtraction of the two yields a discrete "similarity envelope." The problems associated with offsetting surfaces is solved because this discrete representation only deals with whether the part exists or does not exist in a certain location in space. The erosion and dilation algorithms take care of it.

8.1.1.3 Carving and Shaping Tools

The ability to intuitively carve and shape complex freeform surfaces in CAD models is still limited at this time. However, in order to make the design cycle as versatile as possible, a method for carving and shaping should be implemented. Literature shows that there is already research in this area with groups both in the United States and abroad [Salisbury],[Murakami].

One may argue that the physical model should be the only method of input to the design cycle for freeform shapes. This idea is flawed because translation between the two mediums is still the most time consuming task and if little modifications can be made easily in the CAD model then that is where it should be done. The design cycle's main contribution is to allow designers freedom in their work and not constrain them to a certain method.

8.1.2 Wax 3D Printer

As described in chapter 3, work has been done on a drop on demand wax 3D printer using a Tektronics color dye sublimation printhead. In order to explore the speed of the design cycle a working machine should be finished. 3D printing has an incredible advantage over other types of rapid concept modeling techniques in the category of speed because the majority of the material of the part is put down in bulk and bound together. For the wax printer, it may only have to print half to a third of the total volume of the object. Other rapid concept modeling methods being developed right now must build objects one entire element at a time meaning that all the material of the part must be deposited serially. 3D printing will always be two to three times faster than other competitors.

8.1.3 Color Printing

The Tektronics printhead was developed for color printing paper documents and the output of these printers is quite remarkable. Other than just printing objects, there are applications for being able to print with multiple colors. Therefore, when the wax 3D printer is developed, color capability should also be included.

One application for multiple color printing involves the design cycle and another is in the medical field. Being able to print with multiple colors would allow a designer to create an object with internal limitations. An example would be a toy that had a battery pack, transmitter, servo, and wires etc. that were contained on the inside of the part. A blocked out encasement for the parts could be made and the whole object could be printed in different colors as shown in Figure 8-3. The designer can then carve the object and know what depth is permissible.

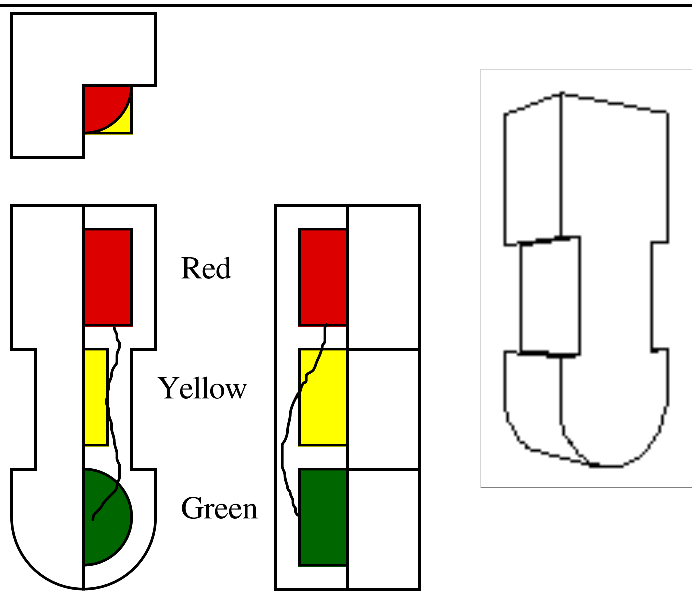


Figure 8-3: Color Encasement

Physicians at the Brigham and Women's Hospital expressed interest in color 3D printing as well. As described also in the applications chapter on page 88, medical visualization using high-end computer graphics is an important tool for doctors. Computers allow them to see and diagnose problems that were previously impossible. However, just like engineers, doctors sometimes need physical models to help them understand a problem.

Color printing in particular, gives the ability for models of a patients body to be created which highlight certain items. For example, a surgeon might want to have a local model of their patients tumor. The tumor and surrounding nerves, bone, and fascia would all be different colors. This gives the surgeon the ability to cut, feel, explore, and see the situation physically as well as with computer graphics before the patient is opened up.

Orthopedic surgeons commented that even with advanced computer graphics, it is difficult to visualize a severe bone fracture or joint rupture. They often don't find out what they need to do until the patient is on the operating table. Using color again to highlight bone, a model could be made and the surgeon could understand a complex fracture beforehand.

8.1.4 Automated Digitizer

Along with building a wax printer, the digitizing process could also be automated. This will help in reducing the translation time between physical and computer model. As a prototype digitizing system, a handheld digital scanner could be mounted onto a CNC milling machine and connected to a PC in order to collect the data. However, this is only a small step above the hand milling and scanning done thus far. Since the digitizing process is repetitive and simple, a new machine that can mill and scan could also be designed and built.

8.1.5 Export for Manufacture

One point that was never formally addressed was that at some point a finished design must leave the cycle and be manufactured either as a single piece or in mass production. The design cycle is a process for designing geometry. Since there is only one material used for the prototyping in this research, it is not an actual manufacturing method.

3D printing is a versatile manufacturing method. It is well established as a good process for ceramic casting shells and research is progressing in tooling for injection molding. If either of these are the method for manufacture of the final designed part then the work is there already exists an avenue to leave the design cycle and produce parts. However, if another method is required, then there are limitations. However, by implementing the suggestions in section 8.1.1.1 Shape Features the CAD model can be exported to virtually any other manufacturing method.

8.2 Conclusions

In this work a complete design cycle has been built where a designer has the ability to manipulate either electronic or physical models and can translate between the two. All the problems associated with completing this design cycle have been solved. Simple as well as complex examples have been demonstrated.

In Chapter 1, a question was raised whether or not this design cycle was useful. Hopefully, this thesis has answered that question in the affirmative. Designers often work within and are limited to the capabilities of their available tools. This research has pushed the envelope of those design limitations concerning complex 3D shape design. There is still much work to be done to make the design cycle a fully integrated and fluid design system. Nevertheless, the design cycle concept has the possibility of becoming a contribution as a powerful new design tool.