

## **2.0 Research Objectives and Thesis Organization**

### ***2.1 Problem Definition***

Finding methods to manipulate free form surfaces has been a challenging topic of research for CAD designers since its inception into the area of Computer Aided Geometric Design. Intersection, trimming and filleting are some of the operations that need to be performed on surfaces for successful completion of any solid model. Also, the designer should be able to elicit useful information from the model like its geometric properties for use in further analysis. In order to do this, the model should be mathematically completely defined.

The issue of trimming surfaces is addressed in this research. In particular, emphasis is on geometric trimming since it not only accomplishes the task of trimming a surface along a given trimming curve, but also provides a mathematical identity to the newly formed trimmed patch. Typically, a trimming curve, which is a resultant of the intersection of two surfaces, would be specified on the surface that has to be trimmed. The main concern in using this trimming curve is that its degree is either unknown or too high for any practical application. A solution, therefore, is sought to approximate the trimming curve

by using a set of lower order curves and subsequently, the trimmed surface, by using a set of lower order surface patches. Suitable subdivision techniques have to be employed in order to achieve this. The subdivision technique should be chosen such that it induces least amount of error between the surface that is trimmed and the newly formed trimmed patch.

Displaying the results of the surface trimming operation is another area of interest to designers. A visualization tool, which can faithfully depict the surfaces and the differences between them by using different rendering schemes, is needed for validating the approximation. Also, providing an error measurement scheme, which can give both qualitative and quantitative feedback on the approximation, would take the process of geometric trimming through one complete cycle. Based on the feedback obtained from the above-mentioned tools, the approximation method can be further refined to minimize the errors.

## 2.2 *Research Objectives*

The objective of this research is to

- Trim surfaces and provide a mathematical identity to the trimmed patch. Parametric subdivision is the technique employed to achieve this. The process involves subdivision of the parametric domain that defines the trimmed portion of a surface. The new set of parameter values obtained upon subdivision is then mapped to get data points lying within the trimmed portion of the surface in model space.
- Optimize this data point set for the trimmed patch further, by filtering only those points that meet certain curvature criterion, and use those points for describing the trimmed patch.
- Develop a tool kit for visually displaying the results.
- Perform an error analysis on the resulting surfaces using a tool developed in *MATLAB* to determine the effectiveness of the approach.

The steps involved in the procedure are:

1. Create a B-spline surface to interpolate a given set of data points.
2. Define the trimming curve in the parametric domain of the surface. The trimming curve can be a product, for instance, of the intersection of two surfaces. The

- trimming curve divides the parametric domain of the B-spline surface into “trimmed” and “untrimmed” regions.
3. Subdivide the “untrimmed” region of the parameter domain based on a suitable subdivision technique.
  4. Employ a mapping algorithm to map points obtained from subdivision of the parametric domain, onto the surface in model space. This will result in a data point set for the “untrimmed” portion of the original B-spline surface.
  5. Use this data point set to interrogate the surface and find points on it where curvature minima or maxima occur. Use the newly found set of points from this surface interrogation to describe a new trimmed surface patch. The new data point set is a subset of the data point set for the “untrimmed” portion obtained from the previous step.
  6. Perform an error analysis on the new trimmed patch to find how closely it matches the original patch.
  7. Visually display the results.

Figures 2.1 to 2.4 provide a sample of the results obtained by following the steps described above for trimming a surface. Figure 2.1 shows a rendered image of the original surface obtained by interpolating a given set of data points. Figure 2.2 is a rendered image of the trimmed patch. Figure 2.3 is a rendered image of the original surface overlapping the trimmed patch. This gives a visual perception of the differences between the two matching surfaces. Figure 2.4 is a plot of the positional errors between

corresponding points on the two surfaces, which quantifies the differences between the two surfaces.

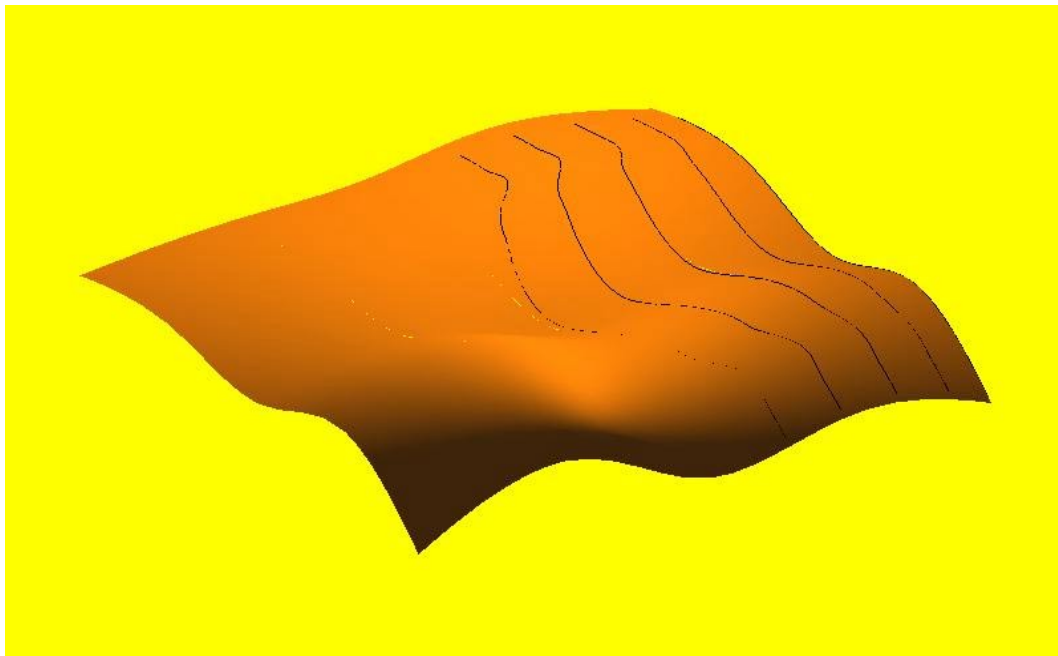


Figure 2.1: A rendered image of the original bi-cubic B-spline surface patch

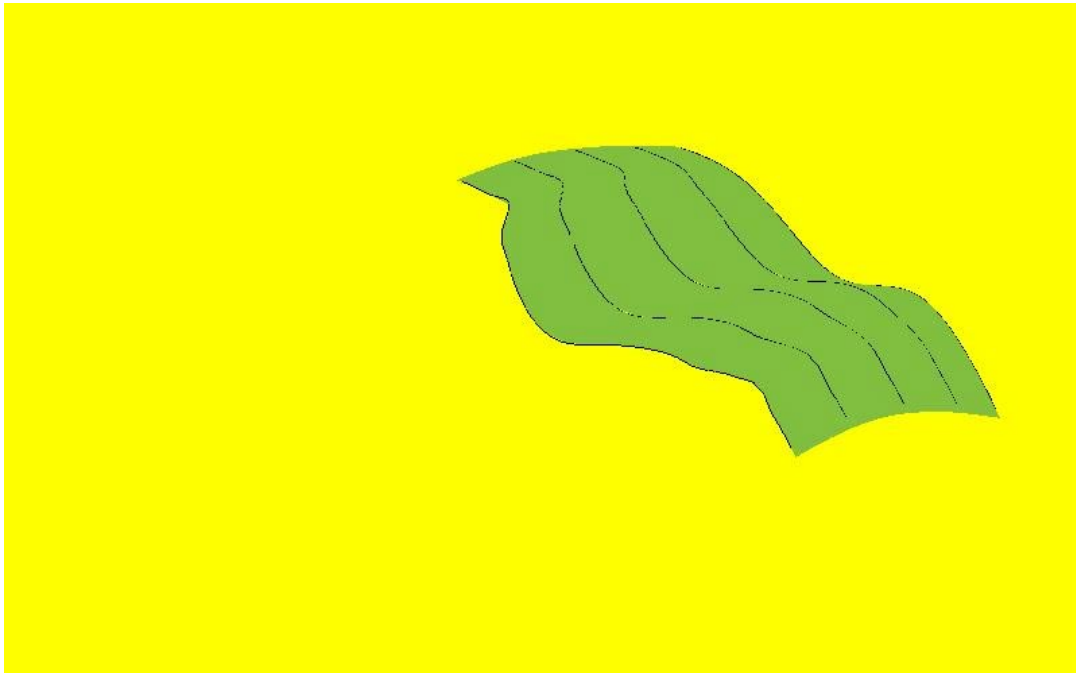


Figure 2.2: Rendered image of the trimmed patch.

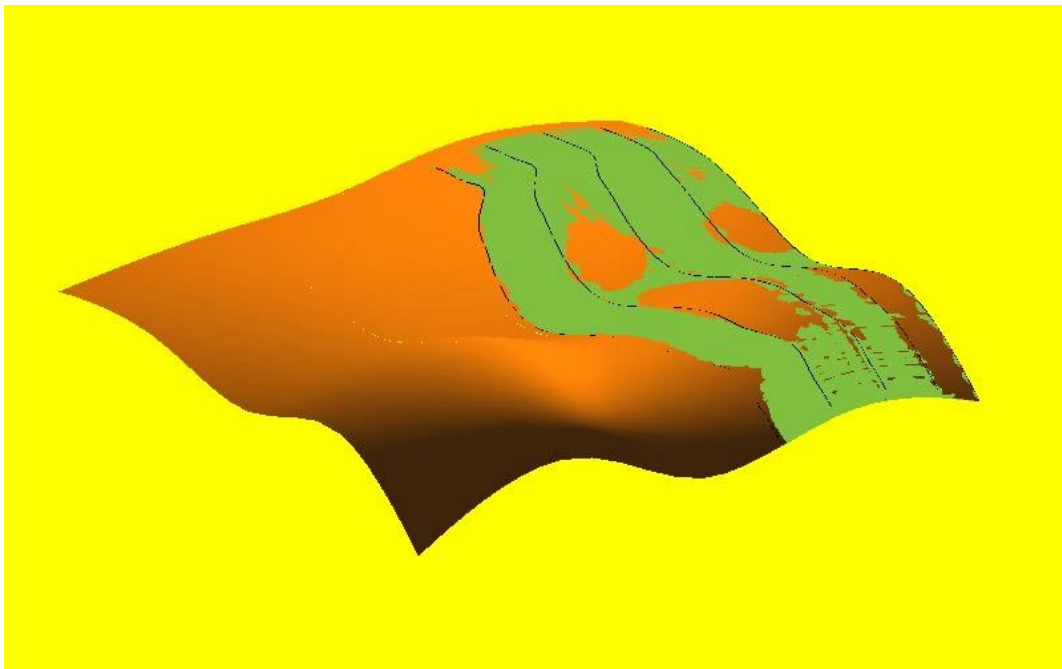


Figure 2.3: Overlapped images of the original and the trimmed patches.

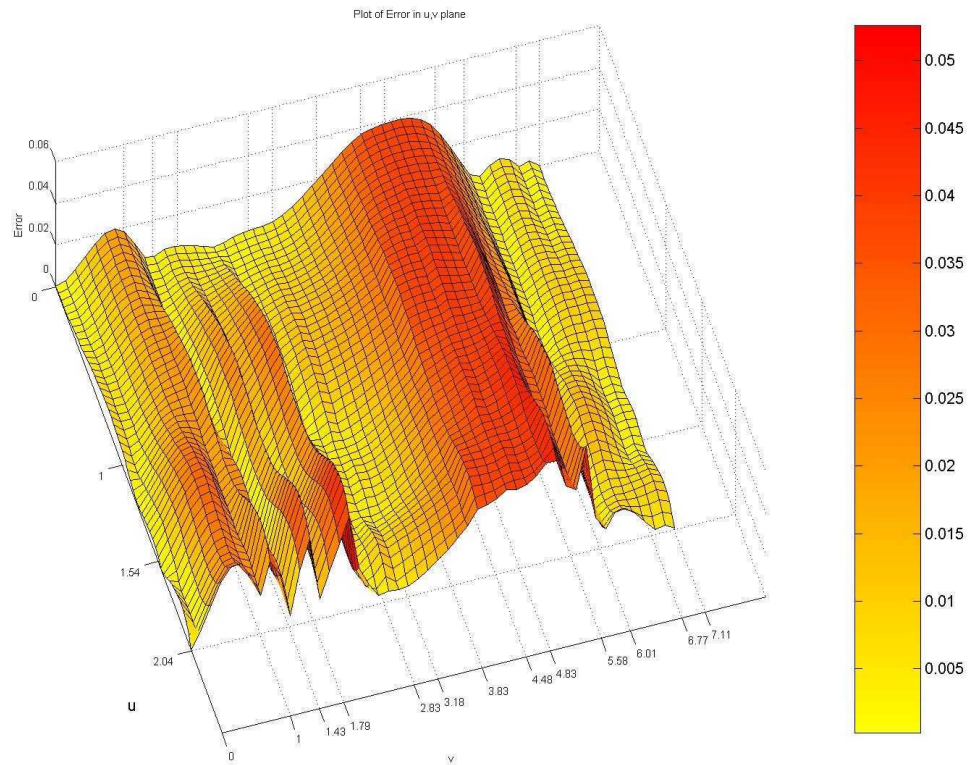


Figure 2.4: Plot of the positional errors between corresponding points on the two surfaces mapped onto the surface's parametric domain. This plot provides with a numerical estimate of the error.



## ***2.3 Thesis Organization***

The method proposed in this research performs a trimming operation on a bi-cubic B-spline surface given a trimming curve. The reader has already been given a brief overview of the various steps involved in this trimming operation. Given below is a brief outline of the topics discussed in this thesis:

1. Introduction
2. Problem Definition, Research Objectives, and Thesis Outline
3. Literature Review
4. Parametric B-splines
5. Geometric Trimming
6. Visualization Tool Kit
7. Results
8. Conclusions

This material is presented with a view to allow readers at different levels to quickly locate areas of interest. The B-splines section gives a brief review of the mathematical representation from a geometric modeling perspective. A description of B-spline differential geometry for curves is also given in this section. The section on Geometric Trimming explains in detail the procedure used to geometrically trim free-form B-spline surfaces. The subsequent section on Visualization describes the OpenGL graphics display

functions that have been used to display and render the B-spline geometry. The implementation of this procedure in the C programming language is included in the appendix along with the data used for the test cases described in this research.