A thesis report on

CUSTOMIZATION OF CAD MODELLING SOFTWARE USING PARAMETRIC MACROS FOR DESIGN OF MACHINABLE ARTISTIC SURFACE PATTERNS

Submitted

In partial fulfillment of the requirements for the award of degree of

Master of Engineering

in

CAD/CAM & ROBOTICS ENGINEERING

Submitted by:

RAHUL SRIVASTAVA

Roll No: 80781021

Under the guidance of

Dr. Ajay Batish
Associate Professor, MED
TU, Patiala-147004

Mr. A. S. Jawanda
Assistant Professor, MED
TU, Patiala-147004

Mr. Ravinder K. Duvedi
Lecturer, MED
TU, Patiala-147004

MECHANICAL ENGINNERING DEPARTMENT

THAPAR UNIVERSITY

PATIALA – 147004

2009

CERTIFICATE

This is certified that the thesis title, "**CUSTOMIZATION OF CAD MODELLING SOFTWARE USING PARAMETRIC MACROS FOR DESIGN OF MACHINABLE ARTISTIC SURFACE PATTERNS**" being submitted by **Mr. RAHUL SRIVASTAVA**, ROLL NO. **80781021**, in partial fulfillment of the requirements for the award of degree of **MASTER OF ENGINEERING** in **CAD CAM & ROBOTICS ENGINEERING** at **THAPAR UNIVERSITY , PATIALA** is a bonafide work carried out by him under our guidance and supervision and that no part of this work has been submitted before for the award of any degree in **THAPAR UNIVERSITY** and at any other place.

Dr. Ajay Batish
Associate Professor, MED
Thapar University,
Patiala-147004

Mr. A. S. Jawanda
Assistant Professor, MED
Thapar University,
Patiala-147004

Mr. Ravinder K. Duvedi
Lecturer, MED
Thapar University,
Patiala-147004

Dr. S.K Mohapatra
Professor & Head, MED
Thapar University,
Patiala

Dr. R.K Sharma
Dean of Academic Affair
Thapar University,
Patiala

ACKNOWLEDGEMENT

I express my sincere gratitude to **Dr. Ajay Batish, Associate Professor, Mr. A. S. Jawanda**, **Assistant Professor and Mr. Ravinder K. Duvedi**, **Lecturer Mechanical Engineering Department, Thapar University**, **Patiala,** for their valuable guidance, proper advice and constant encouragement during the course of my thesis work.

I express my deep sense of gratitude to **Mr. S.K Mohapatra, Head of Department, MED** for this inspirational guidance and wholehearted corporation

I do not find enough words with which I can express my feeling of thanks to the entire faculty and staff of **Mechanical Engineering Department**, **Thapar university**, for their help, inspiration and moral support which went a long way successful completion of the work.

(RAHUL SRIVASTAVA)

ABSTRACT

The design of the artistic feature involves various aspects such as analysis, creativity, and development. The craft work cannot be completed without the manual operations because such workmanship is the outcome of the some special skills owned by the artisans or craftsman. The initial canvas or the base parts required for craftwork are produced using some simple hand tools or manual machine tools as in case of wooden handicraft. As a result it takes considerable amount of time to carve the required base geometry of the part on which finally hand carving is done to generate the final product. The final outcome of the craft work is some specialized items and the number of finished pieces is generally very less. Because of this reason most of the craftsmen prepare the rough canvas for their work themselves than going in for some kind of automation solution.

In the present work a attempt has been made to develop three dimensional model of a specific variety of parametric Rangoli and Phulkari pattern. The MACRO has been developed to generalized the parametric patterns and to allow the user to develop these types of design features with minimum requirement of expertise in this field.

Solid modeling systems are design with an API (Application Programming Interface) which form the canvas for writing MACRO. The API of Solidwork solid modeler has been used for this purpose. The API provides the entire tool required to write and test this MACRO. The MACRO issue geometric instruction to the solid modeler based on parameter provided by user. The geometric instructions are then used to form a solid model of 3D design. Thus the MACROS has been created with a view of providing simple numeric parameter input which could be given on the web interface.Thus making the CAD model in the background on a remote server which has solid modelling software loaded. This reduces the cost of design creation by articians working on a web-page, rather it divides the cost of one CAD system amongst all remote users.

The validation of the parameterized design has been done in solidwork®. The artistic patterns created and parameterized are checked for machinability on a 3-axis milling lathe. These parts are pseudo-symmetric sculptured surfaces on a cylindrical base. The parametric variation of the artistic design patterns are checked for machining on PBG2048 CNC milling lathe,which has been discussed in the thesis.

NOMENCLATURE

$P0,\ P\ , T\ *,\ T, X, Y, Z$	:	Coordinate of points
$u\ , v\ , l$	:	Parameter vary from 0 to 1
$rp, rq, r1, r2, r3, r4$	:	Radius
$\Delta x, \Delta y$	:	Change in x and y direction

INDEX

TABLE OF FIGURES

FIGURES PAGE NO.

CHAPTER 1

INTRODUCTION

Throughout history and around the world, humans have been used various artistic feature to embellish their buildings, their tools, their belongings, and themselves these artistic features gives the world perspective and personality at human scale. All these opportunities for decoration still exist, and are joined by a new class of artifacts, which exist only as information

The design of the artistic feature involves various aspects such as analysis, creativity, and development. In the design process, the designers have to deal with all of these aspects in order to balance the beauty and functions of the products. Moreover, creativity of design typically depends on knowledge, experiences, and perceptions of the designers. Meanwhile, they have to consider their designs in term of possibility in production. The overall conclusion was that no avoidable imperfections should be left on a piece of work, yet the quality of handcraft or the marks of the. Furthermore, in the past hundred years, our understanding of the mathematical structure of ornament has flourished. The tools of modern geometry help us make sense of historical ornament and create new designs. And yet, except for a small number of deliberate forays into computer generated ornament, few attempts have been made to examine the fusion of computer graphics, symmetry and artistic design.

The craft work cannot be completed without the manual operations because such workmanship is the outcome of the some special skills owned by the artisans/ craftsman. The initial canvas or the base parts required for craftwork are produced using some simple hand tools or manual machine tools as in case of wooden handicraft. As a result it takes considerable amount of time to carve the required base geometry of the part on which finally hand carving is done to generate the final product. The final outcome of the craft work is some specialized items and the number of finished pieces is generally very less. Because of this reason most of the craftsmen prepare the rough canvas for their work themselves than going in for some kind of automation solution, as the cost of capital invested may not be distributed over the larger number of units.

The challenges such as shorter product life cycles, frequent design revisions, stringent demands on product quality and shorter times to market has imposed few more constraints before we start to think of designing new automation strategies. Globalization has added further challenges for the artisans and craftsman across the world as they are now being forced to produce the better quality product in shorter time span and sell their products at reasonable prices.

Thus there is an urgent need for cost effective automated system for the handicraft workers/ artisans, so that these people would be able to reproduce the identical finished products in smaller time duration , specially a design tool to augment the design created will aid to the capability of craftmen to a greater extent.

1.1 COMPUTER-AIDED DESIGN (CAD)

It involves creating computer models defined by geometrical parameters. These models typically appear on a computer monitor as a three-dimensional representation of a part or a system of parts, which can be readily altered by changing relevant parameters. CAD systems enable designers to view objects under a wide variety of representations and to test these objects by simulating real-world conditions.

CAD is an important industrial art extensively used in many applications, including automotive, shipbuilding, and aerospace industries, industrial and architectural design, prosthetics, and many more. CAD is also widely used to produce computer animation for special effects in movies, advertising, technical manuals. CAD is mainly used for detailed engineering of 3D models and/or 2D drawings of physical components, but it is also used throughout the engineering process from conceptual design and layout of products, through strength and dynamic analysis of assemblies to definition of manufacturing methods of components.

CAD has become an especially important technology within the scope of computer-aided technologies, with benefits such as lower product development costs and a greatly shortened design cycle. CAD enables designers to lay out and develop work on screen, print it out and save it for future editing, saving time on their drawings. The cad system causes the machines to be used nonstop. It's faster than hands or any other relevant form of creating any type of design.. It's more accurate than a hand drawn design.

1.2 LIMITATION OF CAD SYSTEM

1. CAD systems have no means of comprehending real-world concepts, such as the nature of the object being designed or the function that object will serve. CAD systems function by their capacity to codify geometrical concepts. Thus the design process using CAD involves transferring a designer's idea into a formal geometrical model.
2. Special skills and training is required for using state of the art cad/cam technologies
3. Very expensive.

1.3 TYPES OF ARTISTIC PATTERN

The various types of artistic pattern , which are developed uses different type of curves.. These curve can be described mathematically in two types of category.

1.3.1 TYPES OF CURVES

1. Non-Parametric curve
2. Parametric curve

1.3.1.1 NON-PARAMETRIC CURVE

Non parametric equation can be explicit or implicit for a Non parametric curve, The coordinate y and z of a point on the curve are expressed as two separate function of the third coordinate x as the independent variable. The representation of of general three dimensional curve takes form:

$$P = [x \quad y \quad z]T \;=\; [x \quad \int(x) \quad g(x) \;]T$$

These Non parametric patterns developed in the 2-D space defining by coordinates.

1.3.1.2 PARAMETRIC CURVE

In parametric form, each point on a curve is expressed as a function of of a parameter u. The parameter acts as a local coordinate for points on the curve. The parameter equation for a three dimensional curve in spaces takes the following vector form:

$$P(u) = [x \quad y \quad z]T \;=\; [x(u) \quad y(u) \quad z(u)]T \quad ,Umin \leq u \leq Umax$$

The above equation implies that the coordinates of a point on the curve are the components of its position vector.

These both of the above curve together develop different types of patterns given below:

1.3.1 TYPES OF PERIODIC AND NON PERIODIC PATTERN

i. Frieze
ii. Wallpaper group
iii. Islamic star pattern
iv. Braid
v. Rangoli
vi. Phulkari
vii. Celtic knotwork

1.3.2.1 FRIEZE PATTERN

A frieze group [1] is a class of infinite discrete symmetry groups for patterns on a strip (infinitely wide rectangle), hence a class of groups of isometries of the plane, or of a strip. There are seven different frieze groups. The actual symmetry groups within a frieze group are characterized by the smallest translation distance, and, for the frieze groups 4-

7, by a shifting parameter. In the case of symmetry groups in the plane, additional parameters are the direction of the translation vector, and, for the frieze groups 2, 3, 5, 6, and 7, the positioning perpendicular to the translation vector. Thus there are two degrees of freedom for group 1, three for groups 2, 3, and 4, and four for groups 5, 6, and 7. There are seven distinct subgroups (up to scaling and shifting of patterns) in the discrete frieze group generated by a translation, reflection

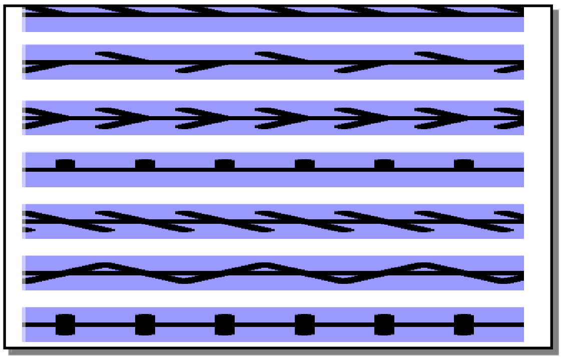


Figure 1.1 Patterns corresponding to the 7 frieze groups

(along the same axis) and a 180° rotation. Each of these subgroups is the symmetry group Without loss of generality, assume the direction of translation symmetry of a frieze pattern is horizontal, the frieze pattern can exhibit five different types of symmetries:

1. horizontal translation,
2. 2-fold rotation (rotation by 180 degrees),
3. horizontal reflection (reflection axis is horizontal),
4. vertical reflection, and
5. horizontal glide-reflection, composed of a half-unit translation horizontally followed by a horizontal reflection.

1.3.2.2 WALLPAPER GROUP

A wallpaper group [2], is a mathematical classification of a two-dimensional repetitive pattern, based on the symmetries in the pattern. Such patterns occur frequently in architecture and decorative art. There are 17 possible distinct groups. The various planar patterns can by classified by the transformation groups that leave them invariant, their symmetry groups. A mathematical analysis of these groups shows that there are exactly 17 different plane symmetry groups.

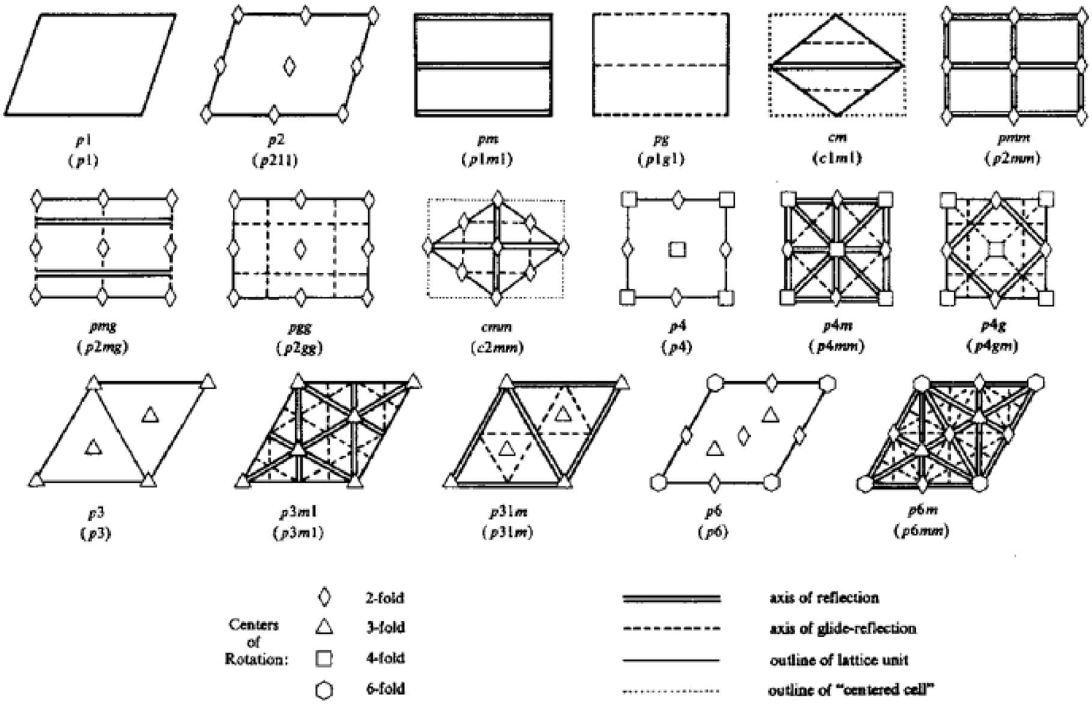


Figure 1.2 The unit lattice for the 17 wallpaper group

1.3.2.3 ISLAMIC STAR PATTERNS

There is no precise definition of an Islamic star pattern [3], but there are some general trends. They tend to be rigidly geometric in design, and feature star-shaped polygonal regions. They can be found carved in wood or stone, built from latticework, or assembled from baked terracotta tiles also style known as *Zelli*.

There are two approach for generating Islamic star pattern

i. Nazm method
ii. Hankin method

The first approach, is the "Najm method", makes use of an explicit understanding of the radially-symmetric motifs that appear in many star patterns. The user chooses specific radially-symmetric motifs for each regular polygon in the template tiling. The computer can then fill the remaining tiles using an "inference algorithm" to create a design consistent with the user's choices. The second approach, is the "Hankin method", does away with explicit motifs and uses only the inference algorithm. In order to obtain some of the characteristic motifs (like rosettes) that are common in star patterns, the Hankin method embeds more intelligence in the template tilings. The interesting question then becomes to compute these new tilings.

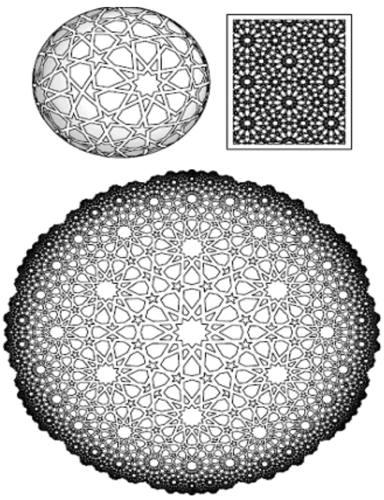


Figure 1.3 Sample of Islamic star pattern

1.3.2.4 BRAID PATTERN

A method of braiding [4] yarns wound on a plurality of carriers arranged in groups as two sets of carriers, the sets moving in opposing directions along continuous, sinusoidal, overlapping paths, such that the spacing of the carriers in a group within at least oneset differs from the spacing of adjacent groups within that set.

Figure 1.4 Braid pattern

1.3.2.5 RANGOLI

Rangoli [5] is the art of drawing images and motifs on the floor and walls of one's home using different color powders. Designed with a beautiful combination of various
colors, the Rangoli images create an enchanting piece of art.

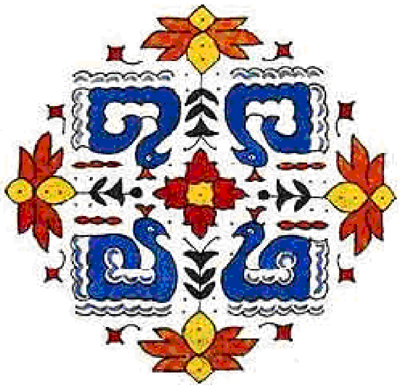


Figure 1.5 Rangoli Pattern

1.3.2.6 CELTIC KNOTWORK

With an instance of Celtic knotwork [6], it is important to distinguish between the abstract design and the style used to render that design. The design in this case is just a network of curves. The rendering styles are numerous; many ideas may be extracted from study of the examples in Bain's book [7] The interlacing of the ribbons endows the picture with an infinitesimal thickness, allowing ribbons to pass over and under each other. Most importantly, the ribbons can be interlaced, cut at crossings to suggest that one ribbon is passing over or under the other. In almost all finished designs, the choice is made so that every ribbon passes alternately over and under at successive crossings

Figure 1.6 Different styles of Celtic knots

CHAPTER 2

APPLICATION PROGRAMMING INTERFACE

2.1 MACRO (Merge and Correlated Recorded Output)

MACRO has been derived from the Greek word 'μάκρο', which stands for long or far in computer science stands for rule or pattern that specifies how a certain input sequence should be mapped to an output sequence according to a defined procedure. The interaction with the model environment is important to improve the design automation process. CAD systems are providing interfaces via Application Programming Interface (API)’s, which enables the user to interact with the model environment with the help of MACROS. Also, with the increasing need for sharing data across applications from various disciplines to provide solutions for real-world problems In many cases there is also the need to implement specific algorithm to perform dedicated an accurate computation which can not be found in any commercial software. These are the main motivations to use Application Programming Interface within a commercial software. This allows to use the predefined native geometrical entities and operations together with an homemade computational algorithm.

2.2 USE OF MACROS

There can be many reasons to write a MACRO. Some of the reasons could be

1. If a particular operation is always done in the same way, every time (or at least most of the time), then MACRO automates common and repetitive tasks by commanding the programmed software to do such tasks.

2. Design optimization: With use of a few inputs from the user, a MACRO can automate the design process weather 2D or 3D designing. Likewise MACRO in general purpose 3D CAD packages can create and analyze design iterations and output the data of each variation.

3. MACRO can from the basis for the development of special tools to help better visualization of the CAD designs.

4. Good MACRO programs can consistently retrieve or export data that already exists in the CAD solid modeler’s documents.

5. Further MACRO increases productivity, save time and money, and helps standardize procedures.

MACRO is generated with the knowledge of some software language. The most commonly used programming languages that one must know before writing MACRO particularly for SolidWorks are listed below.

(i) Visual Basic for Applications (VBA): This application software is installed on the computer when we install SolidWorks (for SolidWorks 2003 and later). Here the MACRO are saved as a single *.swp file. Before writing MACRO in API of SolidWorks one must launch VBA editor from within SolidWorks, to create/edit SolidWorks MACRO.

(ii) Programming languages for writing the stand alone applications: For writing the stand-alone applications, one must be familiar with programming environment like Visual C++.Net, Visual Basic.Net, Visual C++, Visual Basic, Visual Studio. These programming languages have built-in compilers.

2.3 PACKAGE SUPPORTING API

1. SOLIDWORK
2. UNIGRAPHICS
3. PROENGINEER
4. CATIA
5. MECHANICAL DESKTOP

2.4 BENEFIT OF MACRO

Using the software without API'tool MACRO the user can only access to single model entity and the direct access to internal database is not permitted . Using API the database of entity can be directly accessed saving time to execute command and model entities can be interlaced with math and user defined ones. By automating common and repetitive task, it reduces the time consuming in doing a process again and again. So it help laymen to create predefined object with use of graphic interface without having much experience with such cad modeler to create particular type of design.

2.5 MACRO IN SOLIDWORKS

The API uses application of Visual Basic Application to develop procedural algorithm into SOLIDWORK modeler. This allows to use the predefined native geometrical entities and operations together with an *home made* computational algorithm. three kinds of objects: those coming from Solidworks (model native entities), those coming from math utility database (math entities) and user defined entities (Figure 2.1). The native geometrical objects concern the sketch entities (point, line, circle, spline, etc.) and their constraints, the features (extrusion, revolution, loft, etc.), the assembly management (mating, inserting, moving, etc.). The math native objects concern points, vectors and transformations for manipulate entities (projecting from model space to sketch space and vice versa, performing basic operation on vectors, etc.) as shown in Figure 2.1

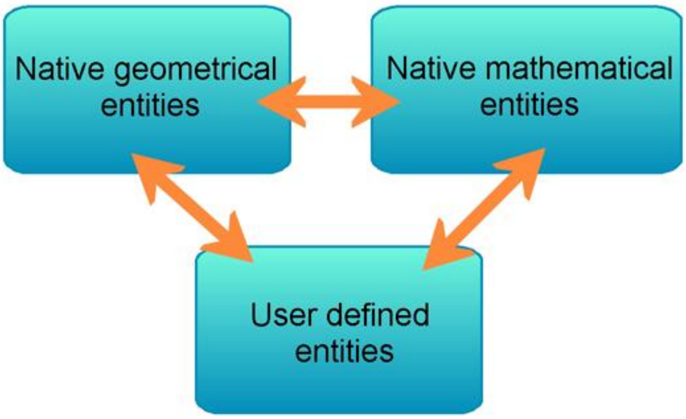


Figure 2.1 Solidworks API entity scheme

Using the software without API the user can only access to single model entity and the direct
access to internal database is not permitted [8] Using API the database of entity can be directly accessed saving time to execute command and model entities can be interlaced with math and user defined ones.

2.5.1 API'S IN SOLID WORK

i. EDrawings API
ii. FeatureWorks API
iii. PDMWorks API
iv. PhotoWorks API
v. PhotoWorks API
vi. SolidWorks Routing API
vii. Solidwork API
viii. SolidWorks Utilities API

2.5.1.1 EDrawings API

The EDrawings application gives you the power to create, view, and share your 3D models and 2D drawings. The eDrawings Application Programming Interface (API) is an OLE programming interface to eDrawings

Use the eDrawings API to:

i. Customize the eDrawings Viewer.
ii. Create interactive web pages.
iii. Translate files.

2.5.1.2 Featureworks API

The FeatureWorks software recognizes features on an imported solid body in a SolidWorks part document. Recognized features are the same as features that you create using the SolidWorks software. The FeatureWorks Application Programming Interface

(API) is an OLE programming interface to FeatureWorks, which allows you to automate FeatureWorks

2.5.1.3 PDMWorks API

PDMWorks is a project-data management application designed to run both natively inside of the SolidWorks environment and as a standalone application.The PDMWorks Application Programming Interface (API) is an OLE programming interface to PDMWorks, which allows you to automate and customize PDMWorks.

2.5.1.4 Photoworks API

Using PhotoWorks, you can create photo-realistic images of exceptional quality of SolidWorks models. PhotoWorks:

i. Provides many professional rendering effects.
ii. Is based on the mental ray rendering engine.

The PhotoWorks Application Programming Interface (API) is an OLE programming interface to PhotoWorks, which allows you to automate and customize PhotoWorks These functions provide direct access to the PhotoWorks environment

2.5.1.5 Solidworks routing API

SolidWorks Routing allows you to create a special type of sub-assembly that builds a path of pipes, tubes, or electrical cables between components.

The SolidWorks Routing Application Programming Interface (API)

i. Allows you to automate and customize only SolidWorks electrical routes.
ii. Is an OLE programming interface to only SolidWorks Routing's electrical routing functionality

2.5.1.6 Solidwork API

This online reference guide documents the SolidWorks APPLICATION PROGRAMMING INTERFACE (API), which you can use to automate and customize the SolidWorks software. These functions provide direct access to SolidWorks functionality such as creating a line, inserting an existing part into a part document, or verifying the parameters of a surface.

2.5.1.7 Solidworks utilities API

SolidWorks Utilities contains tools that let you examine the geometry of a solid model and make comparisons to other models

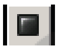
2.6 GENERAL STEP FOR DEVELOPING MACROS

The step require to RECORD macros [9] is given below-

Operations performed is recorded with the SOLIDWORKS user interface and replay them using SOLIDWORKS macros. A MACRO contains calls to the Application

Programming Interface (API) that are equivalent to operations performed in the user interface. A macro can record mouse clicks, menu choices and keystrokes
The macro toolbar contains shortcuts to the macro recording commands. These commands can also be accessed from the **TOOLS, MACRO** menu as shown in figure 2.2

Figure 2.2 Macro toolbar

1. Run  launches the run macro dialog box, where the user selects the macro to execute
2. Stop  launches the save as button macro diolog box where the user enters a valid name and extension for the macro. If the save is cancelled, a prompt appears that allows the user to continue or cancel the recording.
3. Record/pause  allows the user to begin or pause macro recording.
4. New  performs three steps, it launches the new macro dialog box where the user enters a valid name for the macro. Next the macro file is populated with standard lines of programming code to connect to SOLIDWORKS. Finally the VBA editer opens the macro file for the user to begin programming
5. EDIT  LAUNCHES THE EDIT MACRO dialog box for selecting the macro to view or modify.
6. Custom  allows the user to customize a button on the Macro toolbar.An image is selected, a file path to a macro is set, and the custom button is dragged onto the Macro toolbar for use.

The MACRO is recorded by pressing the record button. The activity which is performed in the SOLIDWORK environment, almost all of them recorded successful. Now by pressing the SAVE AS button that macro has been saved successfully. Now that recorded macro has been opened by pressing edit macro button. When that button has been pressed, browse the file saved earlier. The VB editer open the coding of that MACRO. The recorded sample code for the generation of cylinder of radius 50 mm and 100 mm length using MACROS in VB editer is shown below.

```
Dim swApp As Object
Dim Part As Object
Dim SelMgr As Object
Dim boolstatus As Boolean
Dim longstatus As Long, longwarnings As Long
Dim Feature As Object
```

```
Sub main()

Set swApp = Application.SldWorks

Set Part = swApp.ActiveDoc
Set SelMgr = Part.SelectionManager
boolstatus = Part.Extension.SelectByID2("Right Plane", "PLANE", 0, 0,
0, True, 0, Nothing, 0)
boolstatus = Part.Extension.SelectByID2("Top Plane", "PLANE", 0, 0, 0,
True, 0, Nothing, 0)
boolstatus = Part.Extension.SelectByID2("Front Plane", "PLANE", 0, 0,
0, True, 0, Nothing, 0)
boolstatus = Part.Extension.SelectByID2("Front Plane", "PLANE", 0, 0,
0, False, 0, Nothing, 0)
Part.UnBlankRefGeom
Part.SketchManager.InsertSketch True
Part.ClearSelection2 True
Dim SkCircle As Object
Set SkCircle = Part.SketchManager.CreateCirclebyradius(0, 0, 0,0.5)
Part.ClearSelection2 True
Part.SketchManager.InsertSketch True
Part.ShowNamedView2 "*Trimetric", 8
Part.ClearSelection2 True
boolstatus = Part.Extension.SelectByID2("Sketch1", "SKETCH", 0, 0, 0,
False, 0, Nothing, 0)
Part.FeatureManager.FeatureExtrusion2 True, False, False, 0, 0, 0.1,
0.01, False, False, False, False, 0.01745329251994, 0.01745329251994,
False, False, False, False, 1, 1, 1, 0, 0, False
Part.SelectionManager.EnableContourSelection = 0

End Sub
```

CHAPTER 3

LITERATURE REVIEW

The history of proportion in art and architecture has been a search for the key to beauty. The architectural and artistic record indicates that a variety of systems of proportion have been used through the ages in an attempt to create beautiful works. Subjective elements have also played a role; here proportions of an object are modified to please the eye through a slow process of evolution. In architecture this process may extend over many generations in the gradual refinement of traditional forms. In painting or sculpture the process may involve selecting the most admired proportions from nature.

3.1 MODELLING OF ARTISTIC PATTERNS

A lot of research has been done in order to developed artistic design. Over the years, many mathematicians and art historians have focused their attention on the mystery of how he different designing patterns were originally constructed. Many techniques have been proposed, and all are successful in various ways.

Panofsky [10] have reported to use a small number of molds or modules over and over rather than fashioning numerous units of disparate size and shape. Once the modules from which to construct a design have been chosen, the various units must be capable of fitting together to make the finished form. The harmony of proportions should be achieved, according to the Renaissance architect Alberti [11] , reported that nothing could be added, diminished or altered except for the worse Finally, any system of proportions must be flexible enough to express the individual creativity of the artist or architect so that the unexpected may be incorporated into the design.

Kaplan et al [12] have developed a research in the area of computer-generated geometric art and ornament. They have developed a collection of tools and methods for producing traditional Islamic star patterns. He has examined the tessellations of M. C. Escher, developing an “Escherization” algorithm that can derive novel Escher-like tessellations of the plane from arbitrary user-supplied shapes. Throughout, showing how modern mathematics, algorithms, and technology can be applied to the study of these ornamental styles

Bourgoin [13] have created one of the first European collections of Islamic star patterns**.** His book serves as a valuable set of examples for artists and mathematicians. Each pattern has a small section that appears to be inscribed with construction lines. One should not attempt to read too much into these lines. If anything, they are indications of

Bourgoin's transcription process, guidelines he discovered while tackling each individual pattern. They do not provide any prescription of how to construct patterns in general.

Dewdney [14] has presented a complete method for constructing designs based on reflecting lines off of a regular arrangement of circles. Although this technique could be used to construct many well known designs, Dewdney admits that he requires many intuitive leaps to arrive at a finished design.

Dispot's recent Arabeske software [15] allows the user to construct star patterns using an approach similar to Dewdney's.

In the book, Cast´era [16] as dveloped a rich technique motivated by the practicalities of working with the clay tiles used in traditional architectural settings. He starts out with a hand-placed "skeleton" of eight-pointed stars and elongated hexagons called "safts" (a reference to the shuttle used in weaving), and fills the remaining space with additional shapes. With carefully chosen skeletons, he is able to create designs of astonishing beauty and complexity. Cast´era imposes no *a priori* restrictions on a design's symmetries, though by the nature of his construction technique he tends to obtain designs with global eightfold symmetry. Cast´era's designs reflect the Moroccan aesthetic of complex patterns centered around a single large star

The idea of using a tiling as a guide to the construction of star patterns is a common thread that ties together the investigations of many scholars. Evidence of such a tiling-based (or at least tilingaware) construction can be found in the centuries-old. Topkapı scroll [17]. In 1925 Hankin [18]have developed Turkish bath where the star patterns on the walls were accompanied by a lightly-drawn polygonal tiling. Wade [19] also elaborates this construction, presenting what known as the "point-joining technique." He specifies that a design should be developed from a tiling by drawing line segments that cross the midpoints of the tiling's edges. Referring to Hankin, Lee [20] mentions the "polygons in contact technique," stating that new star patterns might be constructed by searching for polygonal tesselations.

Bonner [21], an architectural ornamentalist in New Mexico, has devoted considerable time and energy to the classification and generation of Islamic star patterns. In an unpublished manuscript he details his techniques for producing star patterns, using a tiling-based construction technique

Alexander [22] gave an early demonstration in the second SIGGRAPH conference. More recent examples include Kali [23] Gunn [24] have created a unified system that permits the visualization of symmetric designs in Euclidean, elliptic, and hyperbolic geometry. Although Islamic star patterns have been studied by artists and historians for centuries, it is only recently, with the aid of modern algebra and geometry, that a rigorous mathematical treatment of them can be given. Accordingly, many twentieth-century scholars have discussed various analysis and synthesis methods for star patterns.

Gr¨unbaum and Shephard [25] developed a application of group theory to the study of periodic star patterns. They derive a powerful set of mathematical tools for analyzing patterns in terms of symmetry groups and predicting their properties when elaborated over the entire plane. Abas and Salman [26] carry out this group-theoretic analysis on a library of historical designs. In other work, Abas and Salman [27], developed a plausible patterns from the mathematical tools available to the artisans who created them.

Dunham [28] has created ornamental designs in the hyperbolic plane. Recently, he has adapted several well-known Islamic geometric designs to the hyperbolic plane [29] though each design is developed by hand and no star patterns are included.

Evidence of such a tiling-based (or at least tiling-aware) construction can be found in the centuries-old Topkapı scroll [30] wrote of his discovery of a Turkish bath where the star patterns on the walls were accompanied by a lightly-drawn polygonal tiling. Wade [31] elaborates on this construction, presenting what he calls the “point-joining technique.” He specifies that a design should be developed from a tiling by drawing line segments that cross the midpoints of the tiling’s edges. Referring to Hankin, Lee [32] mentions the “polygons-in-contact technique,” stating that new star patterns might be constructed by searching for polygonal tessellations.

Bonner [33] has built a massive collection of star patterns from tilings, and is the creator of Geodazzlers [34] a set of foldable paper polyhedra decorated with star patterns.
Building on the work of Hankin and Lee, Kaplan [35] presents a software tool that carries out a tiling-based construction on a small set of hard-coded Euclidean tilings.

Software specifically geared towards the construction of tilings of the plane has been around for at least twenty years. Chow [36] has developed a very successful FORTRAN program that let the user input the portion of a tile’s boundary that is “independent,” *i.e.*, not constrained to some other part of the boundary through a symmetry of the tiling. The program then filled in the remaining part of the tile and replicated it in the plane.

Tupper’s Tess [37] has traditionally allowed the user to create drawings belonging to the frieze and wallpaper groups, in a manner similar to the Geometry Center’s Kali [38]. Recently, he modified Tess to support a set of tilings directly.
Like TesselMania!, Tess is geared towards pedagogical use. The three programs above are all based on the Heesch types [39], which are in some sense precursors to the isohedral tilings that do not take into account the additional internal symmetries that a prototile may be forced to have. Roughly speaking, each Heesch tiling represents a set of isohedral types related through a hierarchy of increasingly symmetric tile shapes, and so the Heesch tilings are just a coarse-grained view of the isohedral tilings. Schattschneider reproduces a table of the 28 Heesch types in her book [40], Escher created a small number of carved wooden sculptures featuring spherical interpretations of his

tessellations. Using these as a starting point, Yen and S´equin [41] created an "Escher Sphere Construction Kit," a system that allows the user to design ornamental spherical tilings much as one could create Euclidean tilings in the systems above. As an added feature, the tilings they create can be exported to rapid prototyping hardware and constructed as real artifacts, as was done with spherical Islamic star patterns in Section 3.9. They have created many attractive spherical patterns. Some are fabricated in one piece, others tile by tile, assembled by gluing each tile to the surface of a ball.

Programs for generating mazes are everywhere; there is no shortage of programmers who demonstrate their skills with online maze generators. Yet there is very little research or professional design software that deals with maze generation. Even Berg draws a skeleton of lines by hand, scans in the drawing, and adds in the dead ends manually with a vector drawing tool. Walter Pullen, an amateur maze designer, is the author of the fully featured program Daedalus [42] His "Think Labyrinth" website catalogues an enormous variety of maze styles and algorithms for maze generation. It is the most comprehensive resource we know of for computer-generated mazes.

One other piece of software we are aware of is Peatfield's Maze Creator [43].This commercial tool can produce mazes resembling real-world imagery, but it does so in a na¨ıve way. Peatfield approximates the image with a square grid, and constructs a maze in the grid. The results lack the grace and appeal of the freeform designs by Segala and Berg. In 2004, the puzzle company Conceptis Limited introduced a related puzzle they call Maze-a-Pix [44].

Xu and Kaplan have recently demonstrated a technique for drawing abstract geometric mazes based on arrangements of vortices [45].A vortex is a common obfuscating device in maze design in which multiple spiral arms converge on a common centre. The goal of their work is not stylized depiction. Their mazes are abstract and geometric, and although they include ideas for increasing the appeal of their results, the overall aesthetic range is limited.

The art of the Celts was always non-representational and geometric [7] With the arrival of Christianity to their region in the middle of the first millennium C.E. came the development of the distinctive knotwork patterns most strongly associated with the Celts. Scholars of Celtic art agree that the origin of Celtic knotwork is plaitwork, the weave used in basketry [6] Plaitwork is by no means Celtic; it appears in numerous ornamental traditions. The Celts learned to enrich the ornamental possibilities of plaitwork by breaking crossovers and rejoining the bands systematically.

With an instance of Celtic knotwork, it is important to distinguish between the abstract design and the style used to render that design. The design in this case is just a network of curves. The rendering styles are numerous; many ideas may be extracted from study of the examples in Bain's book [6]

In almost all finished designs, the choice is made so that every ribbon passes alternately over and under at successive crossings. Cromwell [46] developed a simple mechanical process by which a wide variety of knotwork designs may be constructed.

In the language of Shubnikov and Koptsik [47], have developed the symmetries of *two-sided bands*, where the plane is said to be *nonpolar*: its front and back can both contain information. Mercat [48] has developed a more flexible technique that can be seen as an extension of Cromwell's

3.2 CUSTOMIZATION OF CAD PACKAGES FOR GENERATION OF 3D PART MODELS

Literature to support the idea of automation of a general purpose CAD package for the generation of CAD model of the artistic feature has been discussed below. The general purpose data exchange file like STL file generated from the customized CAD modeler will from the basis for the NC tool path planner. The following paragraphs represent the description about the most relevant literature related to the development of the CAD model for different applications.

Scharein [49] in his research has examined topological drawing, a new mode of constructing and interacting with mathematical objects in three-dimensional space. In topological drawing, issues such as adjacency and connectedness, which are topological in nature, take precedence over purely geometric issues. Because the domain of application is mathematics, topological drawing is also concerned with the correct representation and display of these objects on a computer. By correctness the author mean that the essential topological features of objects are maintained during interaction.

Kaplan [3] has reported in his paper about research in the area of computer-generated geometric art and ornament. He has developed a collection of tools and methods for producing traditional Islamic star patterns. He has examined the tessellations of M.C. Escher, and has developed an "Escherization" algorithm that can derive novel Escher-like tessellations of the plane from arbitrary user-supplied shapes. He has demonstrated throughout his work that how modern mathematics, algorithms, and technology can be applied to the study of these ornamental styles.

Sen [50] has used the two-dimensional product-delay algorithm for creating artistic graphic designs on a computer. The author has discussed about a three-dimensional approach to graphic design. The approach extends the product-delay algorithm to three dimensions and has been referred to as the ''product-delay-space curve'' algorithm or simply the PDS algorithm. It has been reported that the PDS algorithm can generate a rich variety of interesting geometric patterns. The methodology has been illustrated with a combination of simple sine waves.

Kaplan and Salesin [51] have presented *Najm*, a set of tools built on the axioms of absolute geometry for exploring the design space of Islamic star patterns. Approach makes use of a novel family of tilings, called “inflation tilings,” which are particularly well suited as guides for creating star patterns. Description of a method for creating a parameterized set of motifs that can be used to fill the many regular polygons that comprise these tilings, as well as an algorithm to infer geometry for any irregular polygons that remain. Erasing the underlying tiling and joining together the inferred motifs produces the star patterns. By choice, *Najm* has been build upon the subset of geometry that makes no assumption about the behavior of parallel lines. As a consequence, star patterns created by *Najm* can be designed equally well to fit the Euclidean plane, the hyperbolic plane, or the surface of a sphere.

Xuewen et al [52] have developed a knowledge-based hammer forging design support system using the API of SolidWorks CAD software package. This prototype system is a design guide system, which provides the right knowledge to support the designer, make decision/selection and design the die step by step. Thus the computerized public and private knowledge can be greatly shared and reused, which can help improve the design efficiency, and avoids previous mistakes effectively.

Stamati and Fudos [53] have introduced ByzantineCAD, a parametric CAD system for the design of pierced medieval jewellery, which is an automated parametric system where the design of a piece of jewellery is expressed by a collection of parameters and constraints and the user’s participation in the design process is through the definition of the parameter values. This system provides the user with the capability of designing custom pierced jewellery in an easy-to-use and efficient manner, using a parametric feature-based design concept. The final piece of jewellery is produced by applying a sequence of operations on a number of elementary solids.

Yoshida and Saito [54] have developed strategies to meet highly aesthetic requirements in industrial design and styling. They have proposed a new category of aesthetic curve segments, where to achieve these aesthetic requirements, they have used the curves whose logarithmic curvature histograms (LCH) have been represented by straight lines. Such curves have been named as aesthetic curves. The overall shapes of aesthetic curves are identified depending on the slope of LCH, by imposing specific constraints to the general formula of aesthetic curves. For interactive control, they propose a novel method for drawing an aesthetic curve segment by specifying two endpoints and their tangent vectors. The authors have also revealed several characteristics of aesthetic curve segments.

Nahm and Ishikawa [55] have developed a vision of set-based parametric design (SBPD) that combines the set-based design practice and parametric modeling technique. A design information solid (DIS) model and a preference set-based design (PSD) model are proposed to incorporate the SBPD approach into current 3D-CAD systems. A DIS-model can be regarded as an integrated CAD data structure that enables the persistent association and consistent management of geometric and non-geometric design information with the CAD geometry. A PSD-model is a new set-to- set-mapping approach for the preliminary engineering design.

Chen et al [56] have proposed an integrative 3D modeling method for resolving the difficult problems of modeling of the complex carving surface. The method is reported to provide the modeling strategy more intelligence and agility, and meets the requirement of digital art carving while at the same time achieving integrative 3D modeling of complex 2D curve flower pattern conveniently. The modeling algorithm is proposed in this paper according to traditional handcraft carving process flow that starts from 2D draft. Firstly a complex 2D curve flower pattern is created and optimized. Then, the coupled intersection points and their midpoints between the scan lines and the pattern curves are obtained by a scan algorithm. Finally, the carving model and track are created by a normal distribution function and a distance function. In the modeling process, the technology of delamination, combination and interpolation are applied which make the visual effect, the precision, and the concave and convex varying of the model surfaces more close to the effect of handcraft carving products. The modeling algorithm can realize integrative 3D modeling of a 2D complex curve flower pattern by the method of processing a 2D data matrix, and greatly decrease operation complexity and data storage capacity while at the same time making 3D modeling of complex carving surface convenient and quick.

Gattamelata et al [57] have discussed about an application of Visual Basic Application Programming Interface (API) to develop numerical and procedural algorithms into CAD software. Their work has its focus on Reverse Engineering embedded into SOLIDWORKS. In many Reverse Engineering applications there is the need to remodel the tessellated surface into an editable solid feature, to analyze it and to manipulate it. For this purpose there can be programmed numerical procedures which interact with native geometrical entities in order to improve the modeling capability using automation protocols. The authors have presented an example of API and Solidworks interaction about the acquisition and processing of surfaces acquired by 3D laser scanner. The problem was to acquire the tessellated geometry, build up a parametric editable feature, perform topological analysis and manipulate more fragments to reconstruct a unique entity. The proposed methodology being based on the integration between native geometrical entities in Solidworks and advanced mathematics algorithms about nonlinear optimization, which can be accessed and manipulated by the user using simple graphic windows. The authors have also described how to implement the interaction among these entities, discussing the role of API focusing on limits and capabilities and presenting the proposed algorithms underling the critical points.

A parametric design technique for describing a large class of designs has been developed by Gross [58] In parametric CAD systems, design features are identified and keyed to a number of input variables. Changes in the input values result in variations of the basic design. Based on conventional software technologies, parametric design has been successfully applied in many design domains including architecture and is supported by several commercial CAD packages. The author has reported that the weakness of parametric techniques is the need to predetermine which properties are input parameters to be varied and which are to be derived. Relational modeling is a simple and powerful extension of parametric design that overcomes this weakness. By viewing relations as reversible rather than one-way, any set of properties can be chosen as input parameters. For example, a relational model that calculates the shadow length of a given building can also be used to calculate the building height given a desired shadow length. In exercising a relational model the designer is not limited to a pre-selected set of input variables but can explore and experiment freely with changes in all parts of the model.

3.3 CONCLUSION OF LITERATURE REVIEW

The generation of artistic feature involves various aspects such as analysis, creativity, and development. In the design process, the designers have to deal with all of these aspects in order to balance the beauty and functions of the products. There are several types of artistic design developed by the various scholars. CAD/CAM offers far more interesting propositions to develop these types of design and make than with simply increasing speed and reducing labor. It also offers the opportunity to thoroughly explore and manipulate form from all angles and viewpoints and the ability to make radical alterations to scale and shape. In traditional approach a model is being generated in any CAD/CAM system.

These CAD/CAM system have been customized to accommodate the requirement. This can make complicate and intricate shape with less expertise in this field.

Though much work has been done earlier for CAD model generation but they were all unique, we are making them generalized so that any user could use it with their own requirement of design choice.

CHAPTER 4

PROBLEM DEFINITION, MODELLING AND ANALYSIS

A computer program in Visual basic has been created in APPLICATION PROGRAM INTERFACE of solid work to generate the MACROS for periodic as well as Non periodic design elements for ultimately creating a 3-D artistic design patterns. The following section describe the details of the mathematical modeling used & The user interface forms those appear in the solidwork environment when the MACROS to create pre-designed artistic patterns have been activated.

4.1 PROBLEM DEFINATION

CAD modeling software are now days available with good Application Programming Interface (API) using which one can customized these packages by writing suitable MACROS to create particular design. The design generation using MACROS takes very short time as well as one need not remember all the commands to generate the particular feature. The present work is to create MACROS in API of solidworks , a propriety of Dassault Systèmes SolidWorks Corp. -a subsidiary of Dassault Systèmes, S. A. Vélizy, France for artistic feature generation. The MACROS have been written in visual basic language for certain parametric and non-parametric rangoli and phulkari pattern. These MACROS have been later used for actual design generation on wood using PBG KW 2048 CNC milling lathe. The overall aim of the present work has been to automate the design creation for artwork process .

4.2 NON PARAMETRIC DESIGN ELEMENTS

The mathematical modeling for various non parametric design elements, for which the MACROS has been created in solidworks environment has been discussed below. The basic canvass for all the type of patch listed below is fixed in the plane. The scale factor and the orientation are given by the user itself. The scale factor scale the patch with the center upper right corner. The rotation of the patch along the point is also fixed. The API function used for the scaling and rotating is given by

```
Part.Extension.ScaleOrCopy
Part.Extension.RotateOrCopy
```

1. Circular patch
2. Triangular patch
3. Elliptical patch
4. Rectangular patch
5. Star patch
6. Rangoli patch

4.2.1 Circular patch

The circular patch element in 3-D can be created on any kind of surfaces .The initial 2-D sketch of the circle can be created in two ways in solidworks.

1. Specifying centre and radius of circle

2. Specifying three points on the perimeter of the circle

But out of these option , circle centre & radius has been used to position the circle on the canvass. The coordinates of the outer canvass for each of the listed above patch have been different.

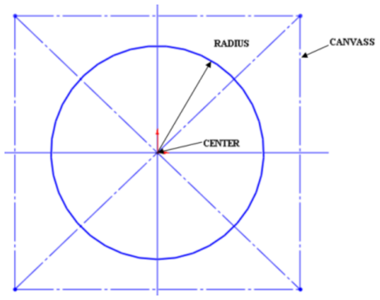


Figure 4.1 Canvass for circular patch in 2D

The scaling factor has to be enter through the user. The range of the scaling factor vary from 0 to 1. Similarly the rotation angle can be vary from 0 to 360° The API commands for the scale factor and rotation angle are as follows.The scaling for the patch is uniform in both axis

4.2.2 Triangular patch

The triangular patch element in 3-D can be created on any kind of surfaces .The initial 2-D sketch of the triangle can be created in two ways in solidworks.

1. Specifying three line interconnected
2. Using polygon option when number of sides is three which can be inscribe or circumscribe a circle

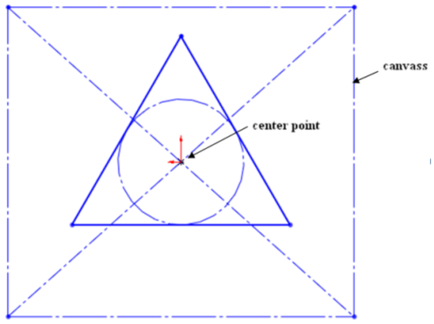


Figure 4.2 Canvass for triangular patch in 2D

Here also scaling factor vary from 0 to 1 scale from upper right corner. The complete patch can be rotated along the center point as shown in figure from 0 to 360°

4.2.3 Elliptical patch

The elliptical entity can be created in 2D sketch by specifying major radius, minor radius and center of point. The canvass can also scale uniformly accordingly user input.

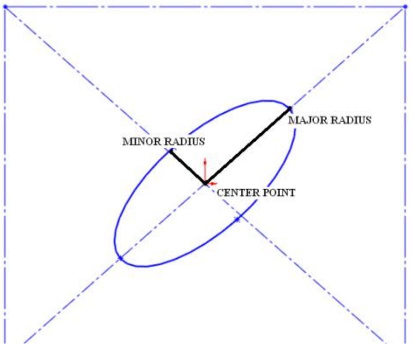


Figure 4.3 Canvass for elliptical patch in 2D

Here also scaling factor vary from 0 to 1 scale from upper right corner. The complete patch can be rotated along the center point as shown in figure from 0 to 360°

4.2.4 Rectangular patch

The rectangular patch element in 3-D can be created on any kind of surfaces .There are various type of method to generate rectangle in 2D sketch.

1. Corner rectangle by specifying left lower and upper right corner
2. Center and upper right corner
3. Three point center rectangle
4. Three point corner rectangle

The center point and upper right corner option has been used in our approach.

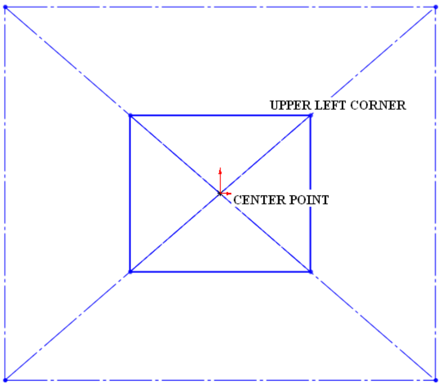


Figure 4.4 Canvass for rectangular patch in 2D

The canvass with rectangular design can be scale up and scale down as user input Here also scaling factor vary from 0 to 1 scale from upper right corner. The complete patch can be rotated along the center point as shown in figure from 0 to 360°

4.2.5 Star patch

The star pattern has been developed by using two triangular entity in the sketch as shown in figure below

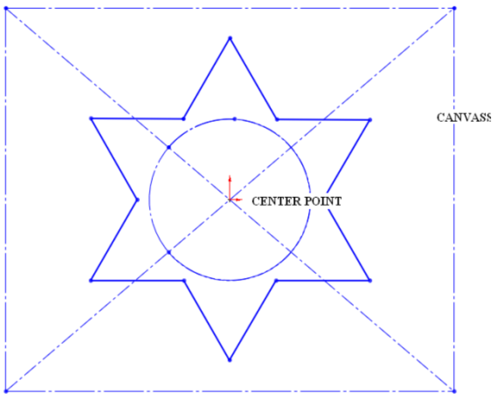


Figure 4.5 Canvass for star patch in 2D

The canvass with star design can be scale up and scale down as user input. Here also scaling factor vary from 0 to 1 scale from upper right corner. The complete patch can be rotated along the center point as shown in figure from 0 to 360°.

4.2.6 Rangoli patch

The basic design of the rangoli pattern is shown in the figure below. The design of the rangoli has been developed using a star entity of 6 sided with the center at C1. The other type of entity used to generate the rangoli pattern is design A1 and design D1. The complete Rangoli pattern has been developed using these two entity by mirroring it along different lines. Design A2, A3 and A4 have been developed by mirroring A1 along line joining C1 and C9, line joining C1 and C10 and line joining C1 and C5 respectively. Similarly design entity D2, D3, D4, D5, D6 have been developed using mirror function

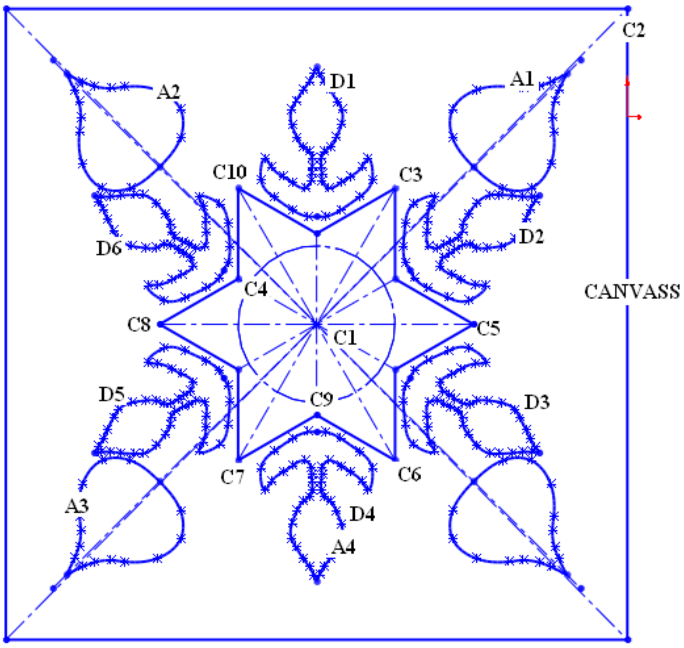


Figure 4.6 Canvass for Rangoli patch in 2D

The complete rangoli design with the canvass can be scale up and scale down as user input which vary from 0 to 6 in our approach. The point from which rangoli have been scale from point C2. The option for the rotation angle for the rangoli pattern have been omitted as it will create a design generated outside the base part.

4.3 PARAMETRIC DESIGN ELEMENTS

The mathematical modeling for parametric design elements, for which the MACROS has been created in solidworks environment has been discussed below. The basic two types of parametric design has been developed in this work is as follows

1. Circular arc pattern
2. Linear line pattern

The initial canvas used for both of these pattern is fixed 2-D sketch. The canvass to develop these type of parametric design became very complicated, that's why there is no option for the scaling factor and the rotation angle have been used as the 3D design may does not evolve for such a complicated canvass.

4.3.1 CIRCULAR ARC PATTERN

In this section, the approach for the parameterization of the design pattern as shown in figure 4.2 and 4.3 has been discussed. The curve A and curve B is parameterized using parameter u whose value changed from 0 to 1. These two curve, one is curve B defined by C5 corresponding to radius r1 and the other one is curve A defined by C4 that is radius r4 of figure 4.2. So when the value of u is changed, this parameter caused the radius of curve B defined by C5 changed from r1 to r2. Similarly the radius of curve A defined by C4 changed from r4 to r3 as shown in figure 4.7 and 4.8

The canvass of the circular arc pattern is defined by points P4, P23, P11 and P19. The size of the canvas is fixed for the circular arc pattern. The curve B defined by C1 have center point at P3, which is fixed. The dotted rectangle defined by points P1, P5, P18 and P14 restrict the curve c1 to go outside and thus design created by curve C2, C3 and C5 together to remain inside the outer boundary. This dotted rectangle is the upper limit of the curve C1 as shown in figure 4.7. Similarly the lower limit of the curve C1 is defined with dotted rectangle created by P6, P8, P16 and P15 as shown in figure 4.8. The u parameter is defined for the curve B defined as C1 varying from 0 to 1 reducing its radius. so the complete pattern defined by u parameter of curve B does not go beyond the upper limit created by P4, P23, P11 and P19 and below the lower limit created by P6, P8, P16 and P15. The varying value of u parameter causes the curve B to change within the lower and upper limit

Similarly the curve B defined by C4 have the center point P2. The center point of the curve B is also fixed. The parameter for changing the radius of the curve B is v. The parameter v also changes from 0 to 1.This curve B also changes from dotted rectangle defined by points P6, P8, P16 and P15. The radius of the design developed by the curve B also increases from lower value to higher value while changing the v parameter. The u and v parameter are also interrelated, thus just changing the value of the parameter u causes the automatically change of v parameter, and resulting design generated by curve

A and curve B also change. The table below describe the various points and connecting arc used in circular arc pattern from figure 4.7 and 4.8.\

Name of entity	Point generating entity
Arc C1	P1, P5 and center point P4
Arc C2	P18, P5 and center point P21
Arc C3	P18, P14 and center point P20
Arc C5	P14, P1 and center point P10
Arc C4	P12, P7 and center point P2
Arc C6	P7, P17 and center point P5
Arc C7	P17, P15 and center point P22
Arc C8	P12, P15 and center point P14
Lower limit of design by arc C4 ,C6, C7and C8	P7, P17, P15 and P12
Upper limit of design by arc C4 ,C6, C7and C8	P11, P4, P19 and P23
Lower limit of design by arc C1, C2, C3 and C5	P6, P13, P16 and P8
Lower limit of design by arc C1, C2, C3 and C5	P1, P14, P18 and P5

Figure 4.7 description of canvass for circular arc pattern

The complete steps of generating this pattern is given as below

- Equation of radius of circle generating curve A and curve B
- Intersection points of both the circles.
- Triangular check for the intersection point
- Calculation of starting and ending point of curve A and curve B

4.3.1.1 EQUATION OF RADIUS OF CIRCLE GENERATING CURVE

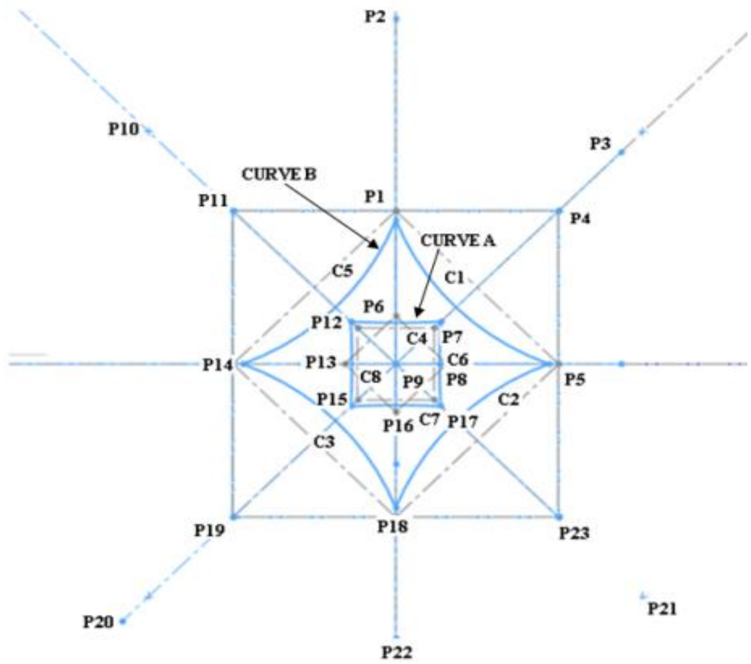


Figure 4.8 Circular arc pattern for u from 0 to 0.5

Both these curve is generated by the circle of different radius. The equation for the radius of the curve B is given by

$$rp = r1 + u * (r2 - r1)$$

Where rp is the change in radius for varying u.When the value of u is 0, it is curve B in figure 4.2 **,**but when value of u is 1 then it is the curve B of figure 4.5.so the governing equation of the circle for the curve B is given by

$$(x - xp3)^2 + (y - yp3)^2 = rp^2$$

Where $xp3$, $yp3$ are coordinates corresponding to point P3.Similarly the equation of circle which is generating the curve A from figure 4.5 to figure 4.4 using parameter v so equation for the radius of the curve A is given by

$$rq = r3 + v * (r4 - r3)$$

Where rq is the change of radius by varying parameter v where v is given by

$$v = 1 - u$$

When the value of v is 0, it is curve A in figure 4.5, but when value of v is 1 then it is the curve A of figure 4.4.so the governing equation of the circle for the curve A is given by

$$(x - xp2)^2 + (y - yp2)^2 = rq^2$$

Where $xp2$, $yp2$ are coordinates corresponding to point P2.So for every value of u there is a value of v and both the parameter u and v develop the equation of circles. Under given constraints, the point P2 and point P3 are fixed.

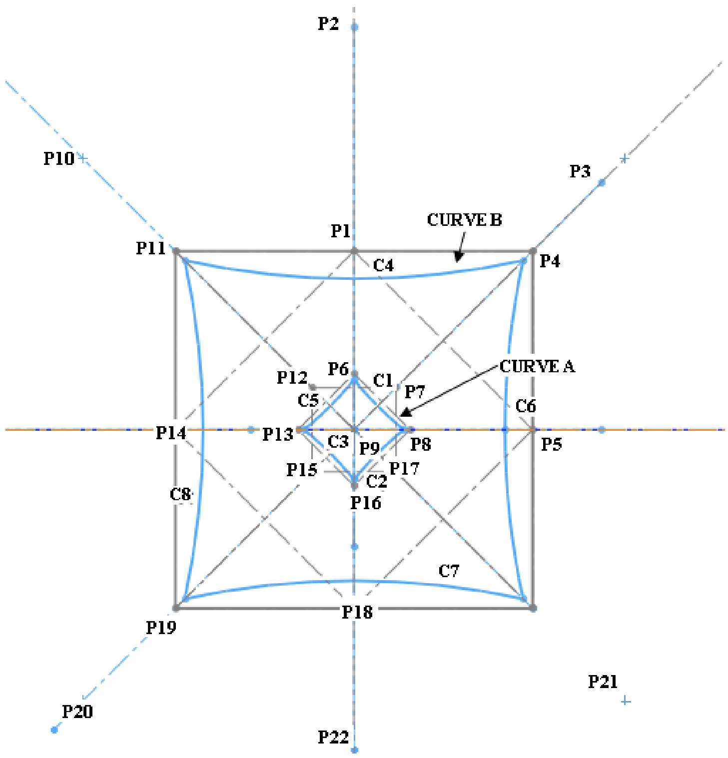


Figure 4.9 Circular arc pattern for u from 0.5 to 1.0

4.3.1.2 INTERSECTION POINTS OF BOTH THE CIRCLES

These two Circles will definitely intersect at two points. Let these points be $(xi0, xi0)$ and $(xi1, xi1)$ one point is greater than point P4 and other one is smaller than point P4 coordinate of both. The smaller intersection point is taken.

$$rp = r1 + u.(r2 - r1)$$

$$rq = r3 + v.(r4 - r3)$$

$$d = \sqrt{(xp3 - xp2)^2 + (yp3 - yp2)^2}$$

$$\mathrm{L} = (\mathrm{rq}^2 - rp^2 + d^2)/\,(2\,.\,\mathrm{d})$$

$$h = \sqrt{rq^2 - l^2}$$

$$xpl = xp2 + l.(xp3 - xp2)\,/\,d$$

$$ypl = yp2 + l.(yp3 - yp2)\,/\,d$$

We have calculated the both intersection points $(zi0, yi0)$ and $(zi1, yi1)$ as given below

$$xi0 = xpl + h * (yp3 - yp2)\,/\,d$$

$yi0 = ypl - h * (xp3 - xp2) / d$

$xi1 = xpl - h * (yp3 - yp2) / d$

$yi1 = ypl + h * (xp3 - xp2) / d$

The smaller of both the point is of concerned. Let the smaller point be $(z0, Y0)$.

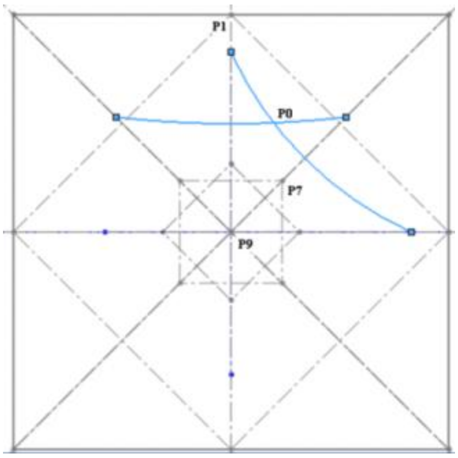


Figure 4.10 Intersection points of both curve

4.3.1.3 TRIANGULAR CHECK FOR THE INTERSECTION POINT

Now intersection point has been checked, whether point lie inside the triangle created by point P1, point P4 and point P9 or not and also above the line $y = y7$ of point P7. If this happens then, there is need to increase the value of u and again we will test the triangular check. Now suppose for any particular value of u, there is value of v, they are not intersecting inside the triangle point P1, point P4 and point P9. The algorithm for triangle check is

$b0 = ((x4 - x1).(y9 - y1) - (x9 - x1).(y4 - y1))$

$b1 = (((x4 - x0).(y9 - y0) - (x9 - x0).(y4 - y0)) / (b0))$

$b2 = (((x9 - x0).(y1 - y0) - (x1 - x0).(y9 - y0)) / (b0))$

$b3 = 1 - b1 - b2$

Now if $b1$, $b2$ and $b3$ is greater than zero and also $y0$ is also greater than $y7$, then there is need to increase the value of u, until they are not intersecting as shown in figure 4.5 below

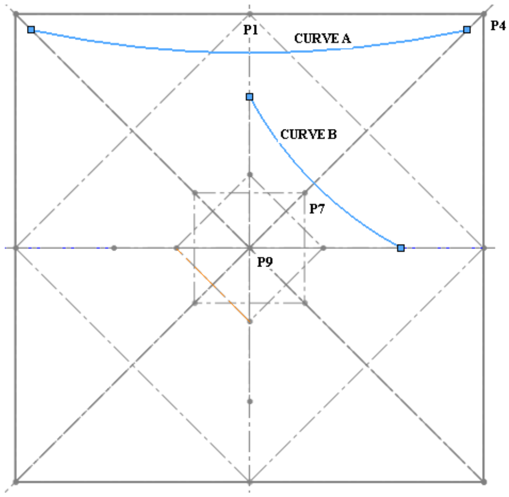


Figure 4.11 Triangular check for the intersection point

4.3.1.4 CALCULATION OF STARTING AND ENDING POINT OF CURVE A AND CURVE B

Now we need to calculate the intersection point of circle generating curve A with $y = 0$ axis and line generated by points P9 and P4.. Similarly intersection point of circle generating curve B with $y = 0$ and $x = 0$ axis. Let the intersection point of curve A with $y = 0$ axis is given by $(xc1, yc1)$ is calculated as

$$xc1 = x1$$

$$yc1 = yp2 - rq$$

The intersection point of curve A with $y = x$ axis is given by

$$mm = 1$$

$$aa = 1 + mm^2$$

$$bb = 2 * ((mm * y9) - (mm * yp2) - (mm^2 * z9) - xp2)$$

$$cc = y9^2 + yp^2 + xp2^2 - rq^2 + (mm * x9)^2 - (2 * mm * y9 * x9) + (2 * yp2 * mm * x9) - (2 * yp2 * y9)$$

$$det = \sqrt{bb^2 - 4 * aa * cc}$$

$$xa1 = ((-bb + det) / (2 * aa * 1))$$

$$xa1 = mm * (za1 - z1) + y9$$

$$xa2 = ((-bb - det) / (2 * aa * 1))$$

$ya2 = mm * (xa2 - z1) + y9$

This is the intersection point of curve A with line $y = x$

$xa = xa2$

$ya = ya2$

The minimum of both the point has been used further.So we get two extreme points on curve A. The intersection point of curve B with with vertical y-axis and horizontal-axis is calculated as below

$av = 1$

$bv = -2 * yp3$

$cv = (yp3^2) - (rp^2) + ((z1 - zp3) * (z1 - zp3))$

$detv = \sqrt{bv^2 - 4 * av * cv}$

$xv1 = x1$

$yv1 = ((-bv - detv) / (2 * av * 1))$

$xv2 = x1$

$yv2 = ((-bv + detv) / (2 * av))$

The above two point $(xv1, yv1)$ and $(xv2, yv2)$ are the intersection point of curve B with vertical y axis as it will intersect with two points on the circle. From above two points, point $(xv1, xv1)$ has been used as it is smaller than other one. Now we will calculate intersection point of curve B with horizontal x axis.

$ah = 1$

$bh = -2 * zp3$

$ch = (yp3^2) - (rp^2) + (zp3^2)$

$deth = \sqrt{bh^2 - 4 * ah * ch}$

$zh1 = ((-bh - deth) / (2 * ah * 1\#))$

$yh1 = 0$

$zh2 = ((-bh + deth) / (2 * ah * 1\#))$

$yh2 = 0$

The above two point $(xh1, yh1)$ and $(xh2, yh2)$ are the intersection point of curve B with vertical y axis. From above two points, the point $(xh1, yh1)$ has been used as it is smaller than other one. So we get two extreme points on curve B

Now the intersection points of the both curve with the y axis and x axis and all the points between them has been calculated and have been mirrored using solidwork's API function for mirroring in 2D plane as shown below.

Part.SketchMirror

4.3.2 LINEAR LINE PATTERN

In this section, the approach for the parameterization of the linear line type pattern as shown in figure below has been discussed. The curve A and curve B is parameterized using parameter u whose value changed from 0 to 1. There are two line segment, one is line B, the other one is line A. So when u parameter changed, the length of both line changed. i.e the length of line A from figure 4.6 decreased to line A in Figure 4.7, simultaneously length of line B increased from Figure 4.10 to Figure 4.11.

The canvass for the linear line pattern is fixed by the points P2, P20, P21 and P22. The line A as defined L5 changes from point P1 and P7 in figure 4.10 up to point P15 and P23 in figure 4.11 by changing parameter u from 0 to 1. The changing value of u parameter changes the line A between lower limit dotted rectangle defined by points P5, P6, P16 and P18 to upper limit rectangle defined by P1, P7, P9 and P10. So the lines A have to remain inside the upper limit rectangle and always above from lower limit rectangle.

Similarly the second line B as shown as L1 is defined by the parameter v which changes from 0 to 1. Here for this line it will change from line defined from points P4 and P19 to points P2 to P22. Here these points provide constraints to the design. This line has been remain inside upper limit points P22 and P2 and remain always above the lower limit points P4 and P19. The table below gives the construction detail of the figure 4.6 and 4.7.

Name of entity	Point generating entity
Line L1	P8, P11
Line L2	P11, P12
Line L3	P12,P13
Line L4	P13,P8
Line L5	P1,P7
Line L6	P7,P9
Line L7	P9,P10
Line L8	P10,P1
Lower limit for design L1, L2, L3 and L4	P19, P4, P14 and P17
Upper limit for design L1, L2, L3 and L4	P22, P2, P20 and P21
Lower limit for design L5, L6, L7 and L8	P6, P16, P18 and P15
Upper limit for design L5, L6, L7 and L8	P1, P7, P9 and P10

Figure 4.12 Description of canvass for linear arc pattern

The steps of generating this pattern is given as below

- Equation of line A and line B of varying length
- Intersection points of both the lines.
- Triangular check for the intersection point

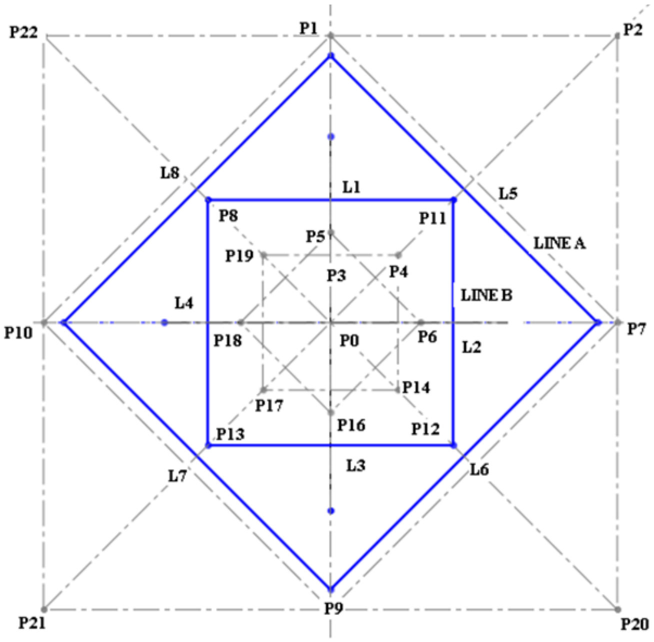


Figure 4.13 Linear line pattern for u from 0.0 to 0.5

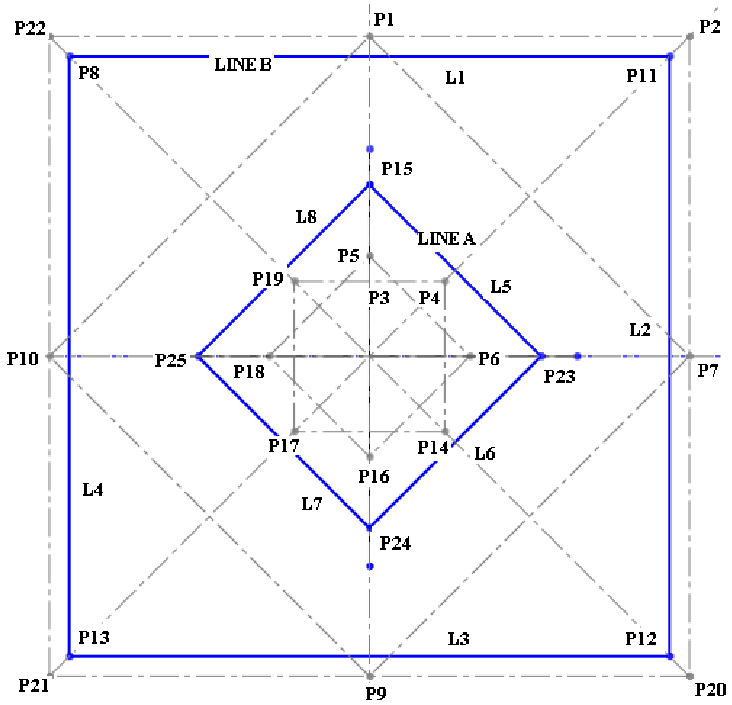


Figure 4.14 Linear line pattern for u from 0.5 to 1.0

4.3.2.1 GOVERNING EQUATION OF LINE A AND LINE B

The parameter governing the equation of the line B is v

$$yk = y3 + v * (y1 - y3) = c1k$$

Where $y3$ and $y1$ are the y coordinate of point P3 and P1. The yk is the point varying with parameter v from 0 to 1

The equation governing line A having parameter u is given by

$$xh = x6 + u * (x7 - x6)$$

$$Yh = 0$$

The point (xh, yh) is corresponding to point 5

$$Yv = y5 + u * (y1 - y5)$$

This point (xv, yv) is corresponding to point 6.The equation of line through point 5 and point 6 is given by

$$yk - yv = ((xv - xh)/(yv - yh)) * (xk - xv)$$

4.3.2.2 INTERSECTION POINTS OF BOTH THE LINES

As the u parameter vary the equation of the line A and B both will vary. There would be certain value of u and correspondingly v, these both line intersect with each other. The intersection point for the line A and line B has been calculated. Let it be be (xk, yk) as shown in Figure 4.12.

The slope of the line A is calculated by

$$mk = (xv - xh)/(yv - yh)$$

$$Yk = mk * zk + (yv - mk * xv)$$

Let $c2k = (yv - mk * xv)$

$$Yk = mk * xk + c2k$$

So $xk = ((c1k - c2k) / mk)$

So the intersection point is given by xk and yk

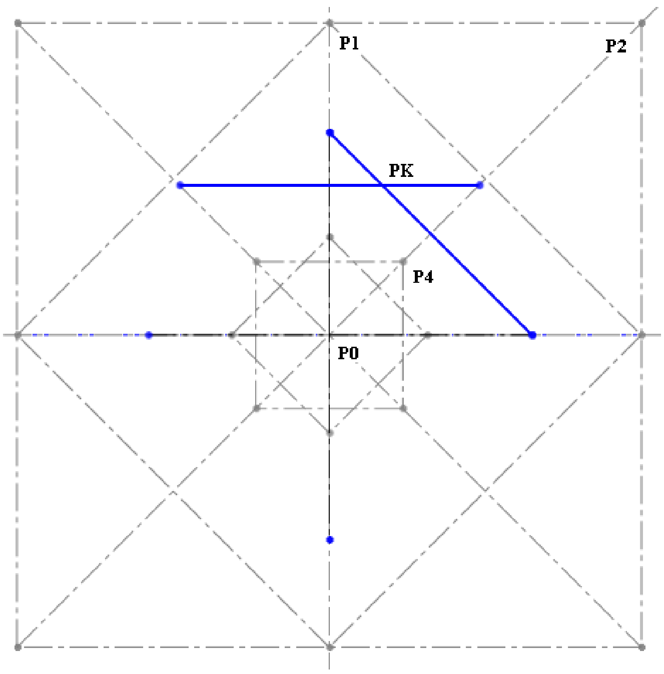


Figure 4.15 Intersection points of both the lines

4.3.2.3 TRIANGULAR CHECK FOR THE INTERSECTION POINT

These two line A and line B will always intersect but there is need to check whether they are intersecting in triangle created by points 1, point 2 and point 0. The triangular check has been used for to check it.

$$bk0 = ((xk1 - xk0) * (yk2 - yk0) - (xk2 - xk0) * (yk1 - yk0))$$

$$bk1 = (((xk1 - xk) * (yk2 - yk) - (xk2 - xk) * (yk1 - yk)) / (bk0 * 1))$$

$$bk2 = (((xk2 - xk) * (yk0 - yk) - (xk0 - xk) * (yk2 - yk)) / (bk0 * 1))$$

$$bk3 = 1 - bk1 - bk2$$

Now if $bk1$, $bk2$ and $bk3$ is greater than zero and also the value of yk is greater than $y4$, then we need to increase the value of u, until they are not intersecting as shown in figure 4.13

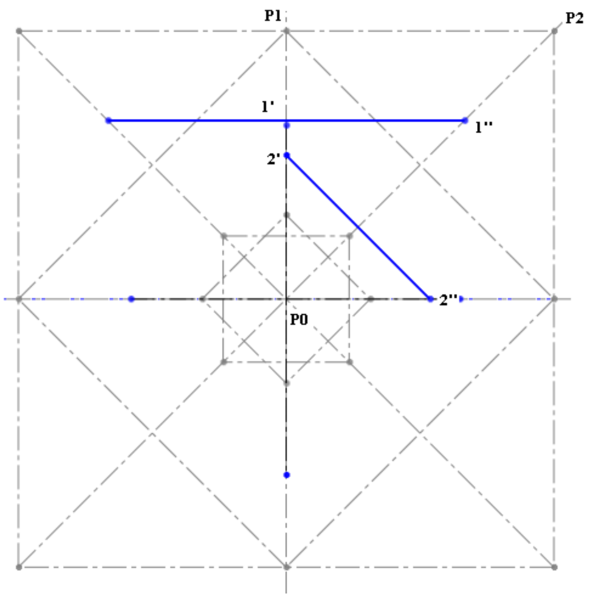


Figure 4.16 Triangular check for the intersection point

There is need to increase the value of parameter u, until they are not intersecting as shown in Figure 4.12

4.3.2.4 CALCULATION OF STARTING AND ENDING POINT OF LINE A AND LINE B

The calculation of starting and end point for the line A and line B is calculated. The coordinate of point 1’ is $(z0, yk)$.The coordinate of 1’’ is calculated as shown in figure 4.12..

$$mb = ((xk0 - xk2) / (yko - yk2))$$

$$x1'' = ((yk - yk0) / mb) + xk0$$

$y1'' = yk$

where mb is the slope of the line segment $0 - 1''$.Similarly the coordinate of point 2' and 2'' is calculated earlier. The point 2' is corresponds to (xv, yv) and the point 2'' is given by (xh, yh).So we get extreme point of line A and line B . Now we can generate line using line equation given by

$$x = zx1 + l * (xt2 - xt1)$$

$$y = yt1 + l * (yt2 - yt1)$$

Where l varies from 0 to 1.For line A, $xt1$ and $yt1$ is xv and yv respectively and $xt2$ and $yt2$ is xh and xv respectively.similarly for line B, $xt1$ and $yt1$ is xv and yk respectively and $xt2$ and $yt2$ is $x1''$ and $y1''$ respectively.
Now the intersection points of the both curve with the y axis and x axis and all the points between them has been calculated and have been mirrored using solidwork's API function for mirroring in 2D plane as shown below.
Part.SketchMirror

4.3.3 MATHEMATICAL MODEL OF RECTANGULAR SPIRAL PATTERN

The rectangular spiral in 2-D has been developed user interective in such a way that user need to enter the number of spirals. The outer rectangular canvas has been kept fixed for this approach.
The equation of the line and the gap between the line has been related in such a way that it makes the number spiral entered by the user. The figure below show the example of patterns.

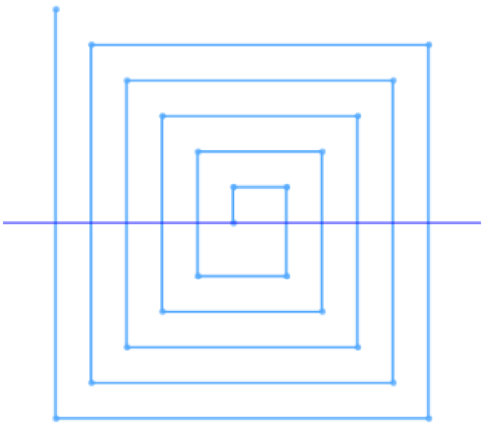


Figure 4.17 Rectangular spiral pattern sketch for no. of spiral 5

4.4 PATTERN DESIGN IN THREE DIMENSION

The various parametric patterns discussed in previous sections can generate two types of three dimensional patterns on the cylindrical part. The software package SOLDWORK is used to develop these 3D patterns on the cylindrical part. These two types of three dimensional patterns are

- Rectangular spiral pattern
- Wrap type pattern
- Sweep type pattern

4.4.1 RECTANGULAR SPIRAL PATTERN

The rectangular spiral pattern has developed in 3D by using sweep feature of circular cross section. The figure below shows the 3D figure for this types of pattern.

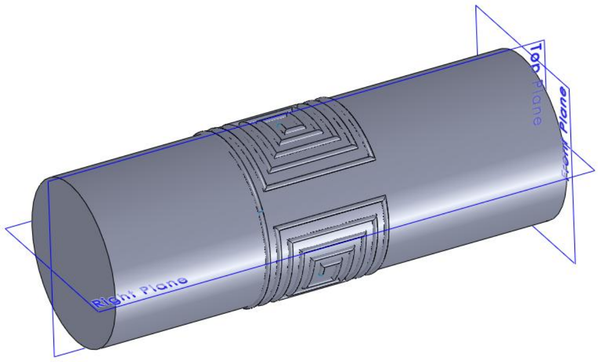


Figure 4.18 Rectangluar spiral pattern

4.4.2 WRAP AND SWEEP PATTERN FOR CIRCULAR ARC AND LINEAR LINE

These patterns are basically developed by the patterns discussed in section 4.2.1, 4.2.2 and 4.2.3. Both these type of pattern also divided into two category

- Pattern with u varying from 0 to 0.5
- Pattern with u varying from 0.5 to 1.0

The different type of wrap and sweep pattern for circular arc and linear line for different value of u has been shown in figure below. been shown in figure

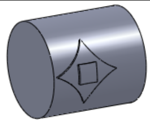
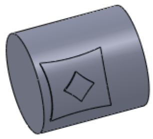
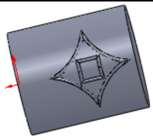
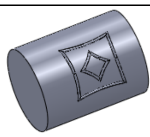
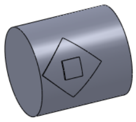
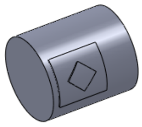
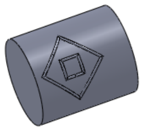
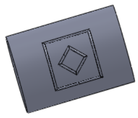
Types of parametric design	Circular arc pattern	Wrap type pattern	U varying from 0 to 0.5	
			U varying from 0 .5 to 1.0	
		Sweep type pattern	U varying from 0 to 0.5	
			U varying from 0 .5 to 1.0	
	Linear line pattern	Wrap type pattern	U varying from 0 to 0.5	
			U varying from 0 .5 to 1.0	
		Sweep type pattern	U varying from 0 to 0.5	
			U varying from 0 .5 to 1.0	

Figure 4.19 Types of different parametric design

4.4 VISUAL INTERFACE FOR MACROS

The price of the general purpose CAD modeling softwares is generally quite high. This makes these packages totally out of reach for the poor artisans especially if we talk in context to the Indian craftsmen. Thus one viable solution that has been thought is to generate the generalized MACRO programs using API (application programming interface) within the CAD modeling softwares to create the three dimensional patterns of some specific variety for the specific user groups. These generalized MACRO programs would be made for the mathematically defined parametric patterns, where just by selecting the value of the control parameter within certain predefined range the design of the 3D pattern would change. The further capability in the MACRO like the options for selecting the shape of base part on which the design pattern is to be generated is to be selected by the user and he would be asked to enter the dimensions of such a base part, around which the selected pattern elements can be wrapped in different orientations. Thereafter, another MACRO would activate with in the solid modeler which would translate the submitted 3D model of the part to suitable neutral format for further processing by the generalized NC tool path planning program. The design developed by the visual basic language to make user interactive programming can be divided into two type. The general layout for the parametric and non parametric design is shown in figure below

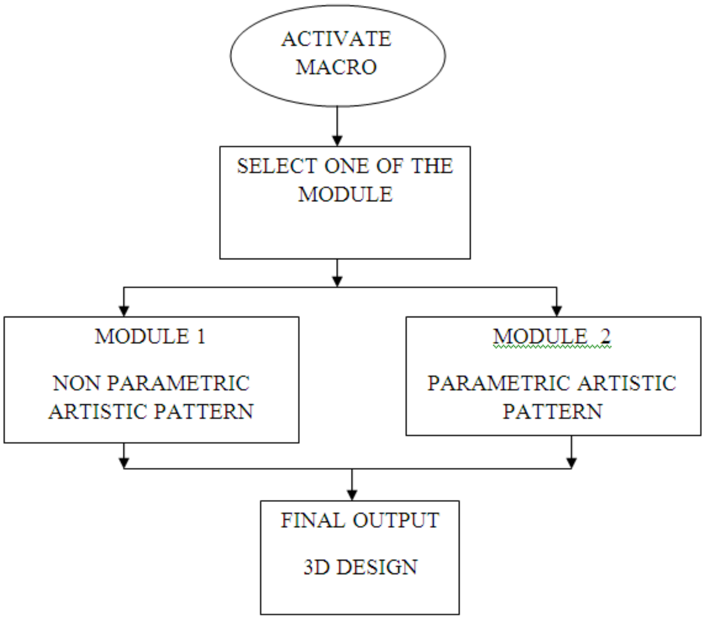


Figure 4.20 Basic layout for design

4.4.1 VB INTERFACE NON PARAMETRIC DESIGN PATTERN

The various types of features developed for non parametric design patterns are as below.

- Circular pattern
- Triangular pattern
- Elliptical pattern
- Rectangular pattern
- Star pattern
- Rangoli pattern

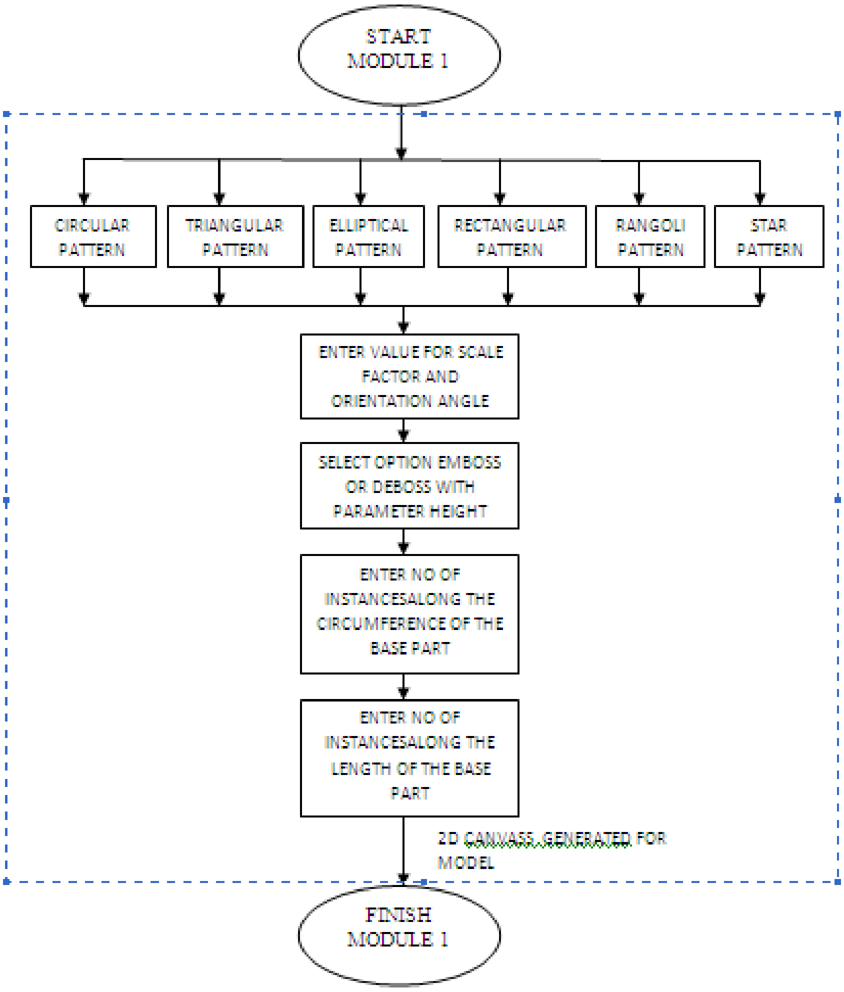


Figure 4.21 General layout for non parametricartistic pattern

The user interface for the non parametric patterns listed above have been developed in the same way , as of the parametric artistic patterns. In the given below section ,the various parameter used to generate non parametric patterns in visual basic interfacing have been discussed. The figures below shows the various entries given by the user to generate non parametric features.

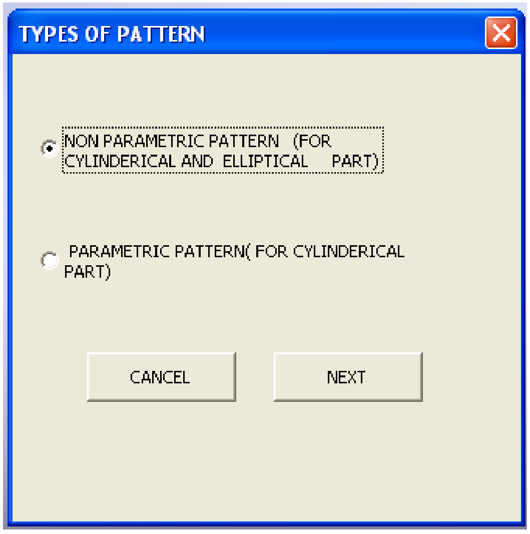


Figure 4.22 Option button to select non parametric design

The user first select the first option button for entering non parametric pattern. These non parametric patterns have been developed for both cylinder and elliptical part of different dimensions.

Now interface wants user to select cylinder or elliptical part as shown in figure below

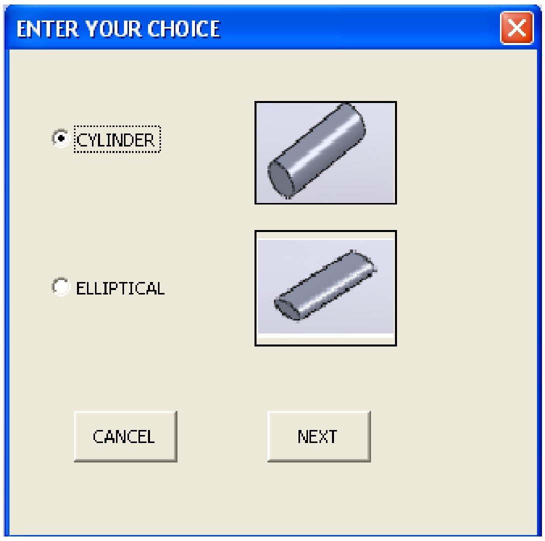


Figure 4.23 Option button to select type of base feature

User can select any of the two options. Suppose it select cylindrical part as base part. Then it will asked for enter dimensions. For cylindrical part it generate following option as shown in figure and for elliptical part it generate following option as shown in figure below.

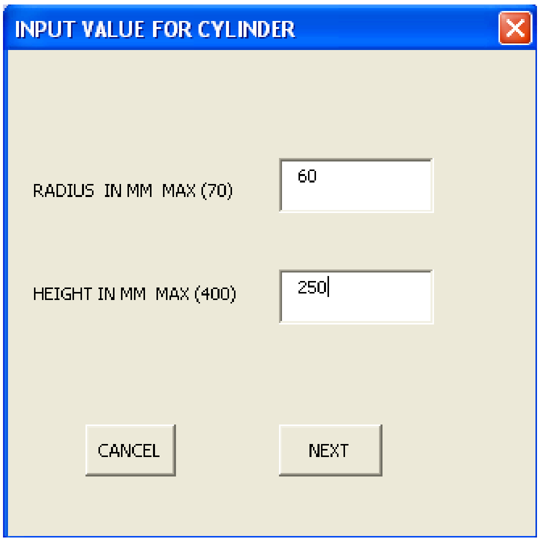


Figure 4.24 Input value for cylindrical parts

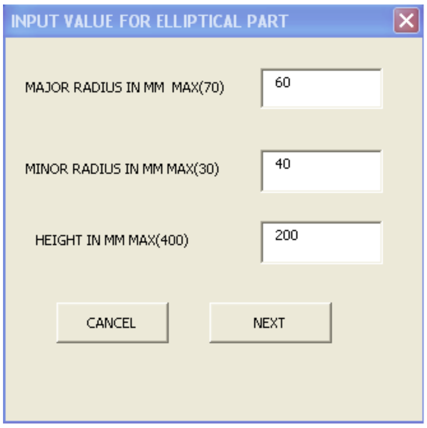


Figure 4.25 Input value for elliptical parts

Now the next option generated is of the type of non parametric designs as listed above as shown in figure below.

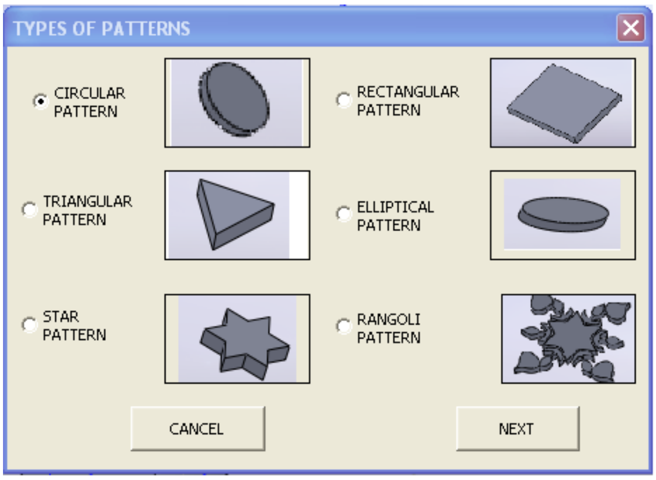


Figure 4.26 Types of non parametric patterns

The user need to enter scale factor and rotation angle for each of the above design as shown in figure below.

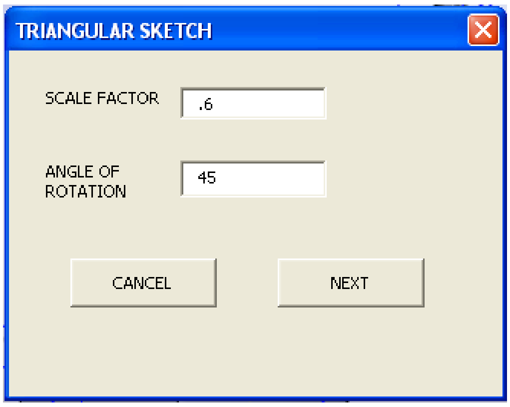


Figure 4.27 Enter value for scale factor and angle of rotation

The next option causes the user to enter the height to be emboss or to be deboss of the particular design on the base feature.

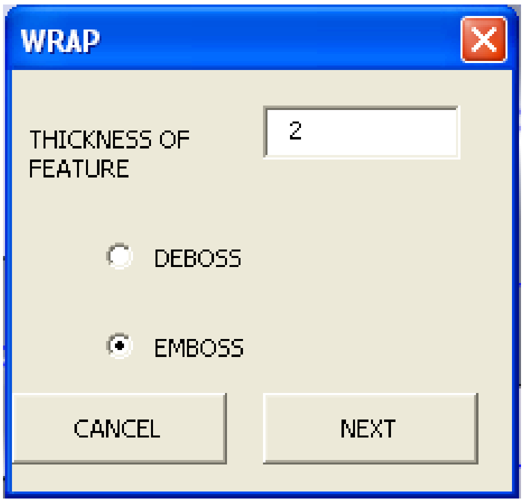


Figure 4.28 Enter height and options for emboss and deboss

The further option provide user to enter the number of instances along the circumference and along the length of the base part. The final part produce the required design.

4.4.2 VB INTERFACE FOR PARAMETRIC DESIGN PATTERN

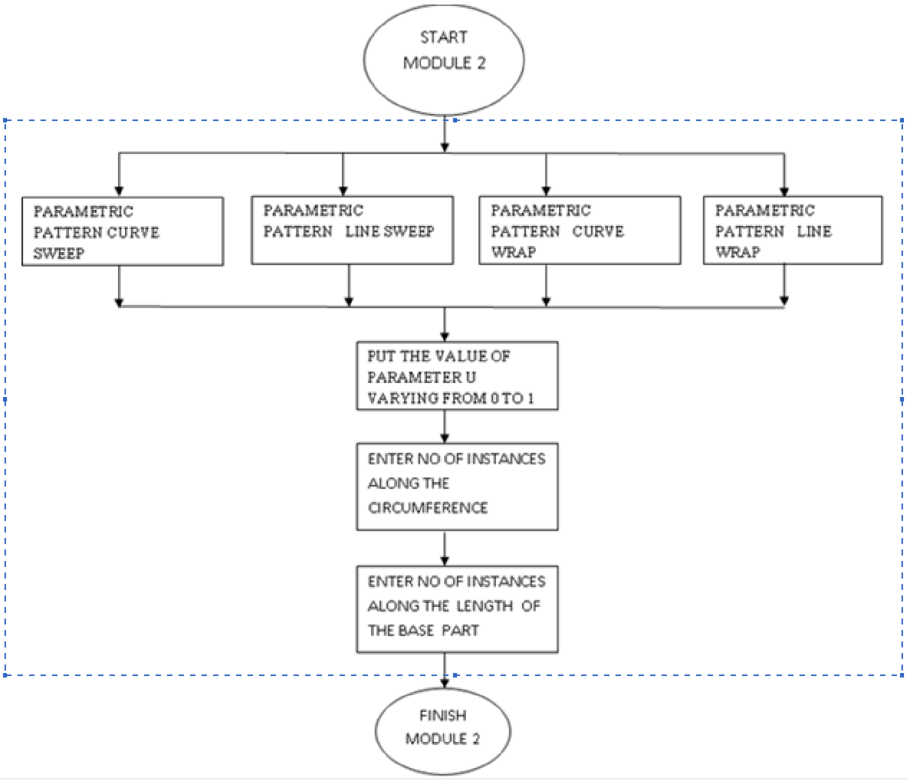


Figure 4.29 General layout for parametric artistic patterns

The parametric patterns discussed in earlier chapter has been developed using MACROS in SOLIDWORK using VBA editer. The VBA editer provide user interface for the solidwork enviroment for designing mathematically defined parametric patterns as shown in figure below.

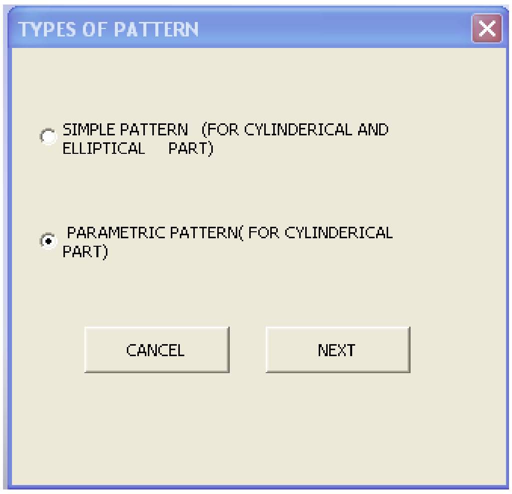


Figure 4.30Option button to select parametric pattern

The user click the one of the option button ,there is no need to design complete design themselves. The user click the next button and the particular design generate itself according the coding provided. Suppose the user click on second option button and click Next. Then it will generate cylinder automatically as shown in figure below.

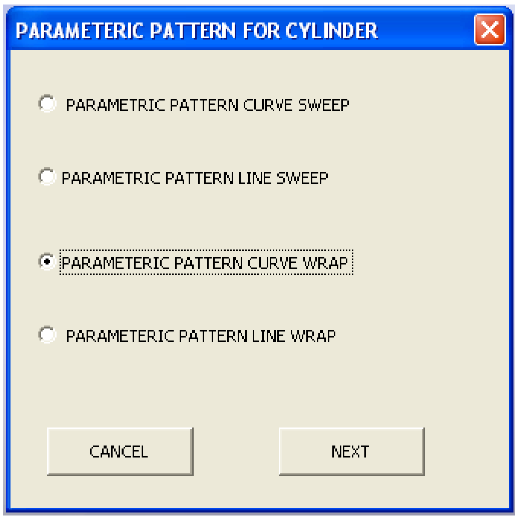


Figure 4.31 Select type of parametric patterns

Now the VBA interface provide further option to the user, to generate which type of parametric pattern. Suppose first option button i.e. parametric pattern curve wrap has been opted by the user. Now the interface ask for the u parameter to enter from the user as discussed in section earlier. suppose user enter u parameter between 0 to 0.5. Let it be 0.3.

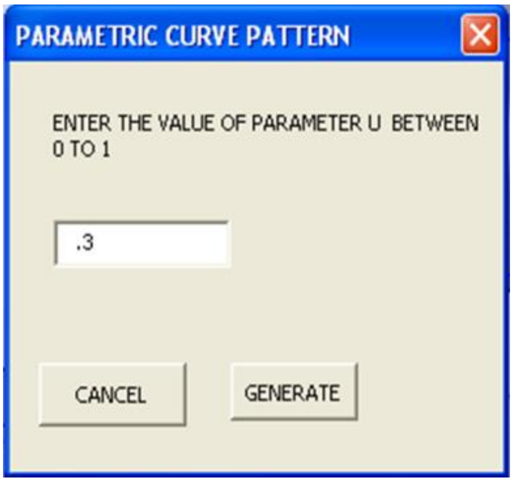


Figure 4.32 Inputbox to enter value of parameter u

It will generate sketch projected on to the cylinder

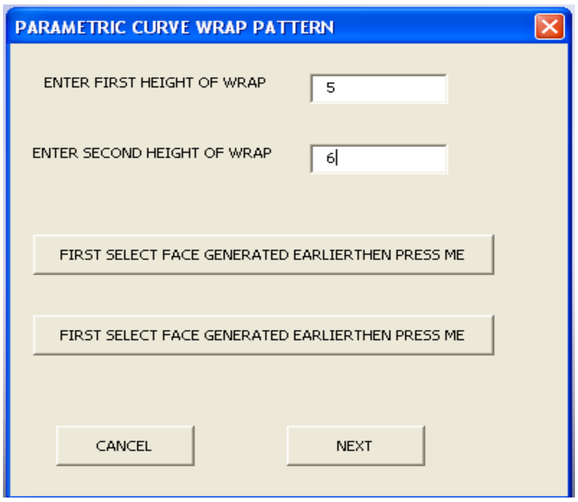


Figure 4.33 Interface for selecting faces and entering depth of height

The interface wants the user to enter the height of both curve from the cylinder. Then it ask to select the both faces one by one. Then the design came will as in figure below.

The interface asked for the number of the patterns along the length, the interface automatically generate the number of instances along the length. It also provide the information to pattern around the circumference of the cylinder. So this will first generate 4 patterns along the length.

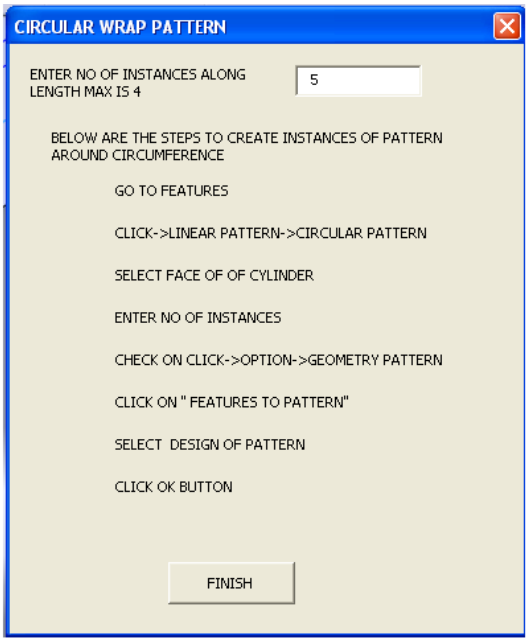


Figure 4.34 Interface to generate number of pattern along axis and circumference

Finally after patterning along the circumference as procedure shown in right side, the design looks as shown in above figure

The final step is to save this design part as .STL file.just click on FILE →SAVE AS TYPE. Click on .STL format. Now click on OPTION button, mark the check box stating "DO NOT TRANSFORM STL DATA TO THE POSITIVE COORDINATE SYSTEM". Now save it in the desired folder.

So the generalized MACRO programs using API (application programming interface) with VBA editer within the CAD modeling software SOLIDWORK create the various types of parametric and non parametric patterns of some specific variety **.** The advantage of such a system is that the end user need not have designing skills or knowledge of using any CAD systems.The user just need to give simple input in vb interface and there it generate the final result in CAD modeler.

CHAPTER 5

RESULTS AND DISCUSSIONS

5.1 VALIDATION

In the present chapter, the design created for non parameter and parametric pattern have been checked for their validation by creating the 3D design element in CAD modeler and the result have been generated as shown in figure below.

The figure belows shows the different types of non parametric pattern developed in Visual Basic interface which have been validated using the EDIT option in CAD modeller.

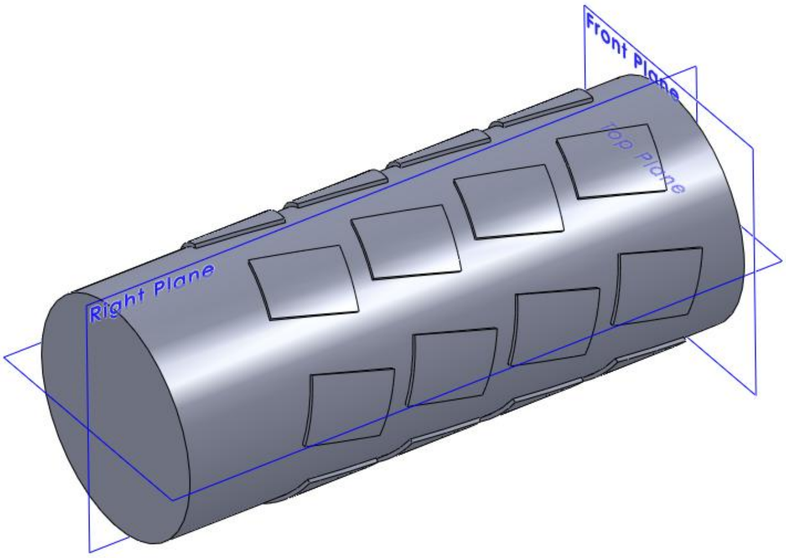


Figure 5.1 Rectangular patterns

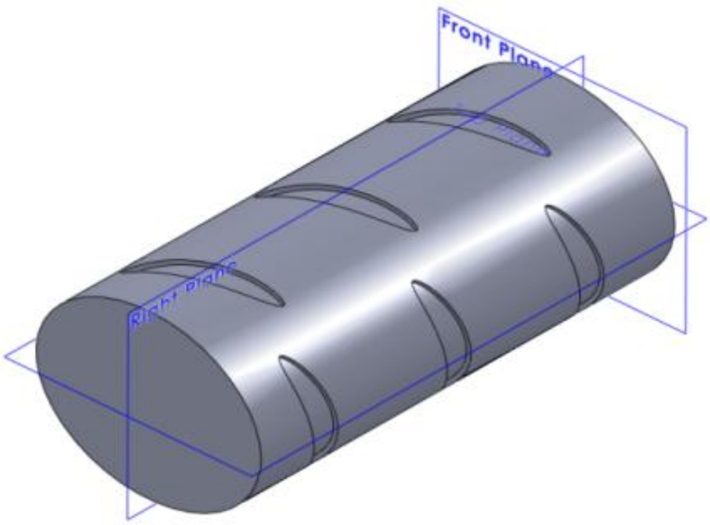


Figure 5.2 Elliptical patterns

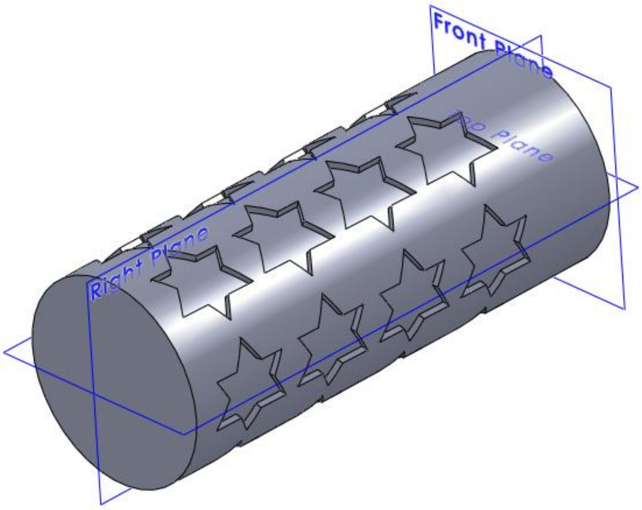


Figure 5.3 Star patterns

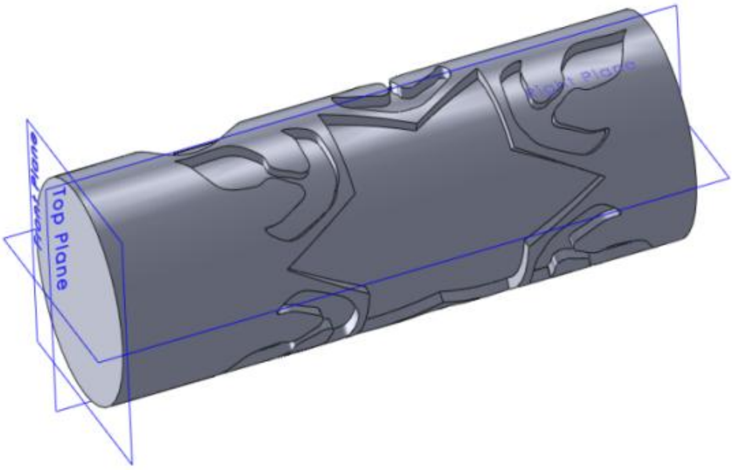


Figure 5.4 Rangoli pattern

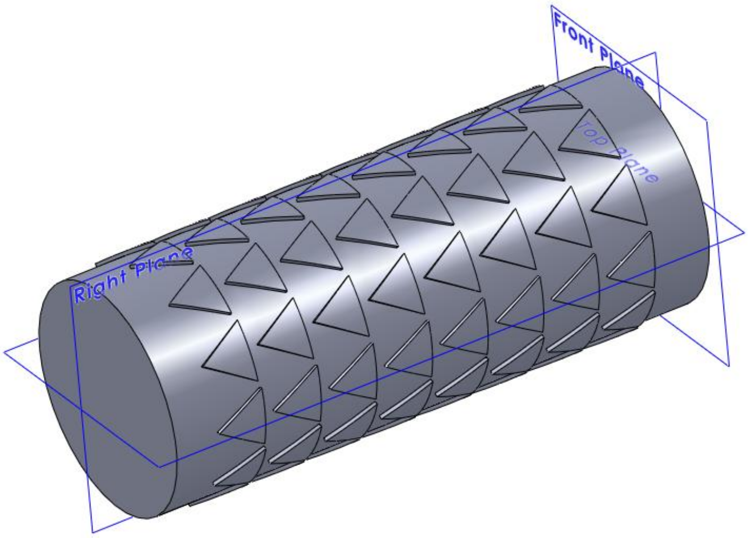


Figure 5.5 Triangular patterns

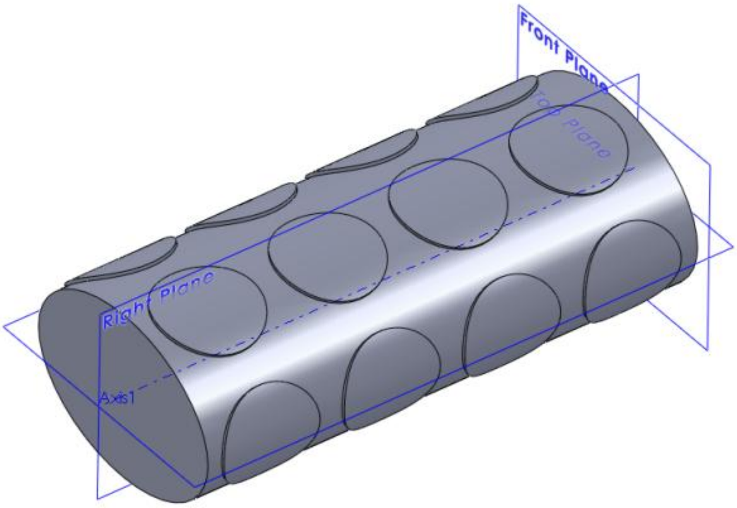


Figure 5.6 circular patterns

The figure from 5.7 to 5.9 shows the various types of design that can be generated using the basic parametric design discussed in chapter 4.

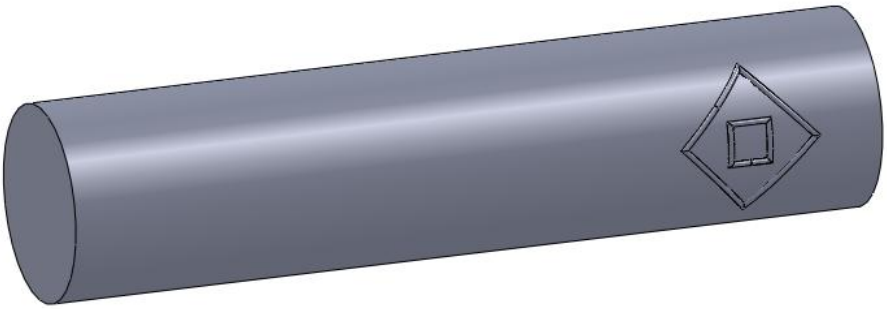


Figure 5.7 Single linear line pattern with u 0.06

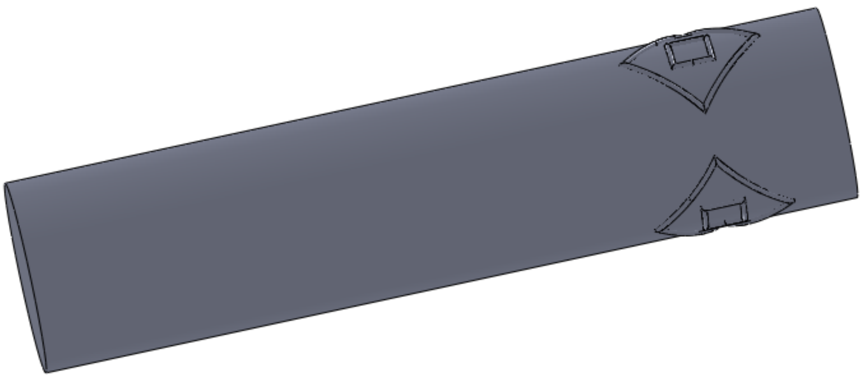


Figure 5.8 pattern of linear line pattern with u 0.06 along circumference

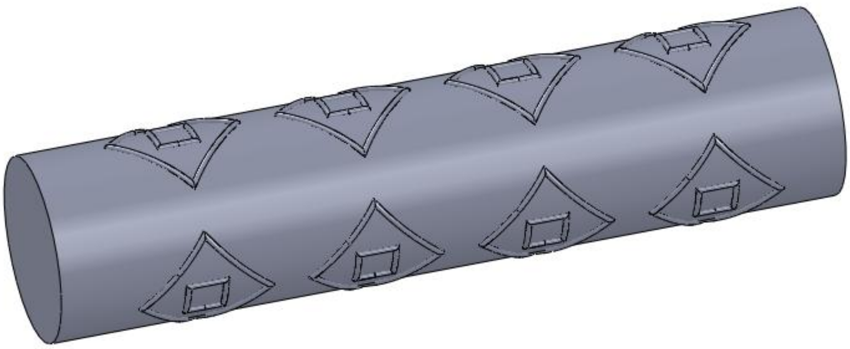


Figure 5.9 pattern of linear line pattern along length without interference

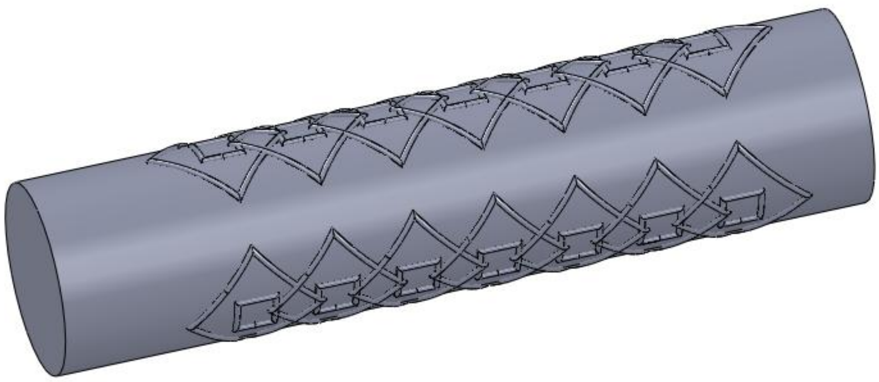


Figure 5.10 pattern of linear line pattern along length with interference

5.2 PHYSICAL REALIZATION OF PARAMETRIC DESIGN OF ORNAMENTAL WOODWORKING CNC LATHE MILL

The various parametric design have been developed on 3-axis surface machining center PBG2048 CNC lathe. The machine operate directly with the computer attached to it. The cutter has the to and fro movement to make contact with the workpiece along the x direction. The workpiece has the rotation along the z direction. The cutter has also movement in y direction to go along the axial length of the workpiece. The motor has controller resolution of 1/2000 in a single turn and the treaded rod has pitch of 20, which translates the Y 1.27 mm per rotation. The stepper motor has resolution of 2000 steps per revolution, thus making the minimum linear movement of 0.000635 mm. The cutter rotates at 28000 rpm under no load conditions. The software installed is Pewin32PRO2 and PBG_EMC. Pewin32PRO2 is used for the controller and other one is for simulating the tool path on the PC monitor.

The manufacturability of the above shown parametric design elements have been checked by machining on this machine. The figure below shows the various patterns developed on this machine.

Figure 5.11 linear line sweep with u 0.78

Figure 5.12 Linear line sweep pattern with u 0.45

Figure 5.13 Circular arc sweep pattern with u 0.89

Figure 5.14 Linear wrap pattern for u 0.75

Figure 5.15 Linear wrap pattern for u 0.35

Figure 5.16 Circular wrap pattern for u 0.70

Figure 5.17 Circular wrap pattern for u 0.35

5.3 CONCLUSION

The basic geometric features required for artisans like the generation of intricate surfaces that can be carved on wood, can be performed with the modern state of the art CAD packages, wherein the various options for change of the design before actual machining is feasible. But it is difficult, rather impossible for the less educated artisans to learn the methods of using the modern CAD packages as such, as a lot of training and engineering skills are required to really make use of such CAD packages for modeling of the intricate shapes. So the generalized MACRO programs using API (application programming interface) within the CAD modeling softwares to create certain artistic parametric patterns of some specific variety for the specific type. These customization reduces labour time to generate this type complicated designs. The artisian need to just know about the interfaces used .

5.4 FUTURE SCOPE

In the present work ,the automation of the generation of particular type of design have been carried out by developing MACROS in API to make it user interactive.The future work can be extend to areas as listed below

There is need to parameterize more of the different type of design generally used as artistic feature

There is also need to completely automate the design process as there is certain area which have been failed now for automating the procee
there is many limitation on the ornamental woodworking cnc lathe mill. There is need to upgrade the machine to make it successful for any type of design.

The web based manufacturing offer large opportunity to make the designs available from anywhere in the world

REFERENCES

[1]. Yanxi Liu and Robert T.Collins, “A Computational Model for Period Pattern Perception Based on Frieze and Wallpaper Groups”, IEEE Transactions On Pattern Analysis And Machine Intelligence, Vol. 26, No. 3, March 2004.

[2]. Yanxi Liu and Robert T. Collins, “A Computational Model for Period Pattern Perception Based on Frieze and Wallpaper Groups”, IEEE Transactions On Pattern Analysis And Machine Intelligence, Vol. 26, No. 4, March 2004.

[3]. Craig S. Kaplan, “computer graphics and geometric ornamental design”, Doctor of Philosophy, University of Washington, 2002.

[4]. Braid pattern, www.wikiepedia.com/wiki/Braid.

[5]. Rangoli pattern, www.wikiepedia.com/wiki/Rangoli.

[6]. George Bain, “The Methods of Construction of Celtic Art” , Dover, 1973.

[7]. Drew Ivan, “The origin and meaning of celtic knotwork”, http://www.craytech.com/drew/ knotwork/knotwork-meaning.html.

[8]. Pezzuti, E., Piscopo, G., Ubertini, A., Valentini, P.P., Milana, M., Di Leginio, R., Una metodologia perl’analisi e l’archiviazione di reperti archeologici basata sul rilievo mediante scanner laser tridimensionali a non contatto, Archiviazione e Restauro di Reperti Archeologici Mediante Tecniche CAD-RP (in italian), Napoli, 2004.

[9]. Mike spens, “Automating Solidworks 2009 using Macros”, SCHROFF development Corporation”, Third edition, 2008.

[10]. Panofsky, E., “Meaning in the Visual Arts. Chicago”, Univ. of Chicago Press, 1955.

[11]. Gadol, J., “Leon Battista Alberti: Universal Man of the Early Renaissance. Chicago”, Univ. of Chicago Press, 1969.

[12]. Craig S. Kaplan, “computer graphics and geometric ornamental design”, Doctor of Philosophy, University of Washington, 2002.

[13]. J. Bourgoin , “ Arabic Geometrical Pattern and Design.” Dover Publications, 1973.

[14]. A.K. Dewdney, “The Tinkertoy Computer and Other Machinations” W. H. Freeman , pages 222–230, 1993.

[15]. Robert Dixon , “ Two conformal mappings in Michele Emmer , editor”, The Visual Mind: Art and Mathematics, pages 45–48, MIT Press, 1993.

[16]. Jean-Marc Cast´era., “Arabesques: Decorative Art in Morocco”, ACR Edition , 1999.

[17]. G¨ulru necipoˇglu, “The Topkapı Scroll — Geometry and Ornament in Islamic Architecture”, The Getty Center for the History of Art and the Humanities, 1995.

[18]. E. H. Hankin, “ The Drawing of Geometric Patterns in Saracenic Ar”t, volume 15 of Memoirs of the Archaeological Society of India , Government of India, 1925.

[19]. David Wade, “Pattern in Islamic Art”, The Overlook Press, 1976.

[20]. A.J. Lee, "Islamic star patterns", Muqarnas, 4:182–197, 1995.
[21]. Jay Francis Bonner, "Islamic Geometric Patterns: Their Historical Development and Traditional Methods of Derivation", Unpublished, 2000.
[22]. Alexander. H, "The computer/plotter and the 17 ornamental design types", In Proceedings of SIGGRAPH, ACM, New York, 160–167 , 1975.
[23]. Amenta,N.and Phillips, M.1996, KALI. http://www.geom.umn.edu/java/Kali/.and Tess Pedagoguery Software Inc. 2000.
[24]. Gunn,C.," Discrete groups and visualization of three-dimensional manifolds" , In Proceedings of SIGGRAPH,ACM, New York, 255–262,1993.
[25]. Gr¨unbaum, B. AND Shephard, G. C, "Interlace patterns in Islamic and Moorish art", Leonardo 25, 331–339, 1992.
[26]. Abas, S. J. AND Salman, "A Symmetries of Islamic Geometrical Patterns", World Scientific, 1995.
[27]. Abas, S.and Salman, "Geometric and group-theoretic methods for computer graphics studies of Islamic symmetric patterns" , Comput. Graph. For. 11, 1, 43–53, 1992.
[28]. Dunham,D.,"Creating hyperbolic escher patterns" 'Elsevier Science Publishers ,241–247, 1986.
[29]. Dunham, D.."Hyperbolic Islamic patterns—A beginning", In Bridges 2001 Proceedings, R. Sarhangi, Ed. 247–254., 2001.
[30]. Necipoˇglu, G., "The Topkapı Scroll—Geometry and Ornament in Islamic Architecture", The Getty Center for the History of Art and the Humanities, 1995.
[31]. Wade, D., "Pattern in Islamic Art", The Overlook Press, 1976.
[32]. Lee.A, "Islamic star patterns". Muqarnas 4, 182–197, 1987.
[33]. Bonner, J. F. , "Islamic Geometric Patterns: Their Historical Development and Traditional Methods of Derivation", Unpublished, . 2000.
[34]. Bonner, J.. Geodazzlers. http://www.dstoys.com/GD.html., 1997.
[35]. Kaplan, C. S. and Salesin, D. H. Escherization, "In Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH 2000", ACM, New York, 499–510, 2000.
[36]. William W. Chow, " Automatic generation of interlocking shapes", Computer Graphics and Image Processing, 9,333–353, 1979.
[37]. Pedagoguery Software Inc. Tess. http://www.peda.com/tess/Welcome.html.
[38]. Nina Amenta and Mark Phillips, Kali. http://www.geom.umn.edu/java/Kali/.
[39]. H. Heesch and O. Kienzle, "Flachenschluss", Springer–Verlag, 1963.
[40]. Doris Schattschneider, "M.C. Escher: Visions of Symmetry", W.H. Freeman, 1990.
[41]. Jane Yen and Carlo S´equin, "Escher sphere construction kit,In Proceedings of the 2001 symposium on Interactive 3D graphics", pages 95–98. ACM Press, 2001.
[42]. Pullen, W. D," Think labyrinth", http://www.astrolog.org/labyrnth/maze.htm, 2005.
[43]. Peatfield, G., "Maze creator", http://www.mazecreator.com/,2005.
[44]. Conceptis Limited, Conceptis puzzles. http://www.conceptispuzzles.com, 2006.

[45]. Xu, J., and Kaplan, C. S, "Vortex maze construction",Journal of Mathematics and the Arts 1, 1 (March), 7–20, 2007.
[46]. Peter R. Cromwell, "Celtic knotwork: Mathematical art, The Mathematical Intelligencer", 15(1):36–47, 1993.
[47]. V. Shubnikov and V. A. Koptsik, "Symmetry in Science and Art", Plenum Press, 1974
[48]. Christian Mercat, Cours d'enluminure `a base d'entrelacs. http://www.bok.net/kri/celte/.
[49]. Robert Glenn Scharein, "Interactive Topological Drawing", Doctor of Philosophy Thesis, The University of British Columbia, March 1998.
[50]. Asok K. Sen, "Three-Dimensional Approach to Graphic Design", Journal of Computers & Graphics, vol. 27, pp. 303–310, 2003.
[51]. Craig S. Kaplan, and David H. Salesin, "Islamic Star Patterns in Absolute Geometry", University of Washington and Microsoft Corporation, ACM Transactions on Graphics, vol. 23, No. 2, pp 97–119, April 2004.
[52]. Chen Xuewen, Zuo Siyu, Chen Jun and Ruan Xueyu, "Research of knowledge-based hammer forging design support system", International Journal Advanced Manufacturing Technology, vol. 27, pp. 25-32, 2005.
[53]. Vasiliki Stamati and Ioannis Fudos, "A parametric feature-based CAD system for reproducing traditional pierced jewellery", Journal of Computer-Aided Design and Applications, vol. 37, pp. 431-449, 2005.
[54]. Norimasa Yoshida and Takafumi Saito, "Interactive aesthetic curve", Journal of Visual Computation, vol. 22, pp. 896–905, 2006.
[55]. Y. E. Nahm, and H. Ishikawa, "A new 3D-CAD system for set-based parametric design", International Journal Advanced Manufacturing Technology, vol. 29, pp. 137–150, 2006.
[56]. Yutuo Chen, Xuli Han, Minoru Okada and Y. Chen, "Integrative 3D Modelling of Complex Carving Surface", Journal of Computer-Aided Design, vol. 40, pp 123-132,2008.
[57]. Gattamelata, Davide, Pezzuti, Eugenio Valentini, and Pier Paolo, "Using Application Programming Interface to Integrate Reverse Engineering Methodologies into SOLIDWORKS", Department of Mechanical Engineering, University of Rome Tor Vergata Via del Politecnico, 1 – 00133, Rome, Italy.
[58]. Mark D. Gross, "Relational Modeling", Dissertation at M.I.T, Design Technology Research, 14 Centre Street, Cambridge Massachusetts 02139.
[59]. Ibrahim zeid, "CAD/CAM Theory and practice", TATA Mc GRAW-HILL EDITION, Twelth reprint, 2004.