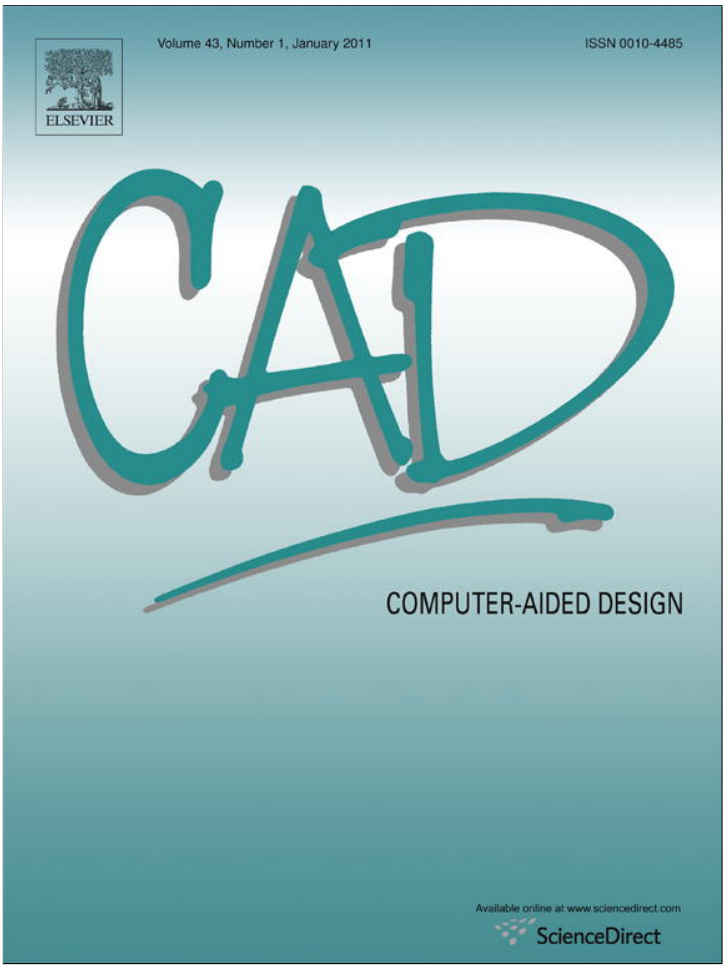


(This is a sample cover image for this issue. The actual cover is not yet available at this time.)

Computer-Aided Design 43 (2011) 1814–1828

Contents lists available at SciVerse ScienceDirect

Computer-Aided Design

journal homepage: www.elsevier.com/locate/cad

A parametric feature-based approach to reconstructing traditional filigree jewelry☆

V. Stamati [a], G. Antonopoulos [a], Ph. Azariadis [b], I. Fudos [a,*]

[a] Department of Computer Science, University of Ioannina, Greece

[b] Department of Product and Systems Design Engineering, University of the Aegean, Greece

ARTICLE INFO

Article history:
Received 23 July 2010
Accepted 6 July 2011

Keywords:
Filigree jewelry
Parametric design
Jewelry patterns
Feature-based reconstruction
Re-engineering
Braids

ABSTRACT

This paper presents a novel approach to reconstructing traditional filigree jewelry. Our method aims at producing an editable CAD representation that can accurately capture the original design and be capable of re-parameterization and modification prior to manufacturing (for example to insert custom designs and abide to free-form artistic alterations). To achieve this, we have developed robust and accurate representations of patterns, used in the design of such jewelry, based on spirals, circular and elliptic arcs, curve segments and braids of various types; all optimized by fairness criteria for aesthetic purposes. We have also built a library of parametric, constraint-based, manufacturable solid patterns that occur frequently in filigree jewelry. For the purposes of this work, a suite of software tools called *ReJCAD* has been developed, that is able to process a highly accurate point cloud of jewelry pieces and to detect features which are fitted by the primitives of the pattern library through user interaction. The point cloud, in the current framework, guides the assembling of all patterns into one robust manufacturable solid piece. We demonstrate the unique capabilities of ReJCAD by reconstructing a filigree brooch part commonly used in late 19th century in northwestern Greece.

1. Introduction

Computer Aided Design (CAD) systems are widely used in most industries and are increasingly used in jewelry manufacturing [1,2]. While manual design of jewelry is still in wide use, this approach is both cumbersome and time consuming when compared to designing using 3D CAD systems.

Editing and redesigning are feasible in a 3D CAD environment, as long as they are supported by parametric and constraint-based techniques [3,4]. 3D rendering helps the artist to detect parts of the model that are unsatisfactory [5]. In addition, it is possible to redesign based on customer feedback after browsing the first version. Generally, current feature-based systems include various tools to assist in designing a piece of jewelry, including constraints, transformations, libraries of jewelry parts and other solid patterns such as cut stones and gems.

Despite the effectiveness of current 3D CAD systems for jewelry, there are categories of jewelry that cannot be reconstructed even with modern CAD systems. Examples of such types of jewelry are traditional pierced jewelry, filigree jewelry and modern jewelry of free-form design. Filigree jewelry is made using a technique of twisting, bending, wrapping and braiding plain precious metal wires (gold and silver) to create a lace-like effect of pierced jewelry (Fig. 1). The metalwork is often combined with precious stones, crystals or glasses to create both jewelry and non-jewelry artifacts. Standard forms of jewelry produced in filigree are earrings, bracelets, brooches, pendants, chains, necklaces and buttons.

Findings in the Valley of the Kings [6] suggest that this technique was first used by ancient Egyptians. Wires of variable diameter were often used either to decorate precious metal surfaces or to craft an entire jewelry piece. From Egypt the filigree technique spread to ancient Greece and Europe and reached Persia and India [7]. In Tibet it was extensively used to decorate the inner parts of the traditional ga'u pendants which the Tibetans used as amulets or charms or credentials of authority [7]. Fig. 2 portraits the use of the filigree technique in such jewelry pieces. The filigree technique flourished and India still remains a massive producer of filigree based jewelry.

Filigree is a technique that produces impressively elaborate artifacts, which combine elegance with the aura of tradition. It became widespread, not only due to its elaborate and yet delicate results, but also because it allows for the creation of maximum size objects by consumption of a minimal amount of precious metal. However it is very delicate, tedious and time-consuming, properties which have forced local Mediterranean craftsmen to

☆ This work was partially supported by Eureka Project #3506, REJCAD under a subcontract for the Hellenic Center of Gold Silversmiths Trade.

* Corresponding author. Tel.: +30 26510 98805; fax: +30 26510 98881.
E-mail address: fudos@cs.uoi.gr (I. Fudos).

0010-4485/$ – see front matter © 2011 Elsevier Ltd. All rights reserved.
doi:10.1016/j.cad.2011.07.002

Fig. 1. A filigree brooch.

refrain from its use. This raised the need for a procedure that would diminish the time consumed and remove the bulk of the labor of this technique.

In this paper, we present a novel approach to reconstructing, creating, and customizing jewelry of filigree craftsmanship (Fig. 1). A suite of CAD tools called *ReJCAD* has been developed, providing the means to the end user to digitally reconstruct, personalize and manufacture filigree-type jewelry using parametric and feature-based techniques. More specifically, this paper makes the following technical contributions:

- Introduces a representation scheme for modeling filigree patterns using elliptical arcs, Bezier segments, spirals and other curve segments. Geometric constraints are imposed on the model in the underlying filigree design library to provide the necessary robustness, editability, and aesthetic conformity to traditional design patterns.
- Describes a novel approach to modeling braids using rational Bezier curve segments which yields an aesthetically improved class of braids.
- Presents a partially automated re-engineering process that aids the reconstruction of a CAD model of traditional filigree jewelry using information derived from a point cloud acquired by high precision 3D scanning.

The remainder of this paper is structured as follows. Section 2 summarizes the current state of art on the open problems that have been tackled in this work. Section 3 presents the proposed classification of parametric constraint-based patterns for creating editable 3D models of filigree jewelry; it also presents a new approach to modeling braids of strands using rational Bezier curves. Section 4 describes the process of re-engineering and reconstructing jewelry pieces, whereas Section 5 presents evaluation results: an example case, a usability evaluation and examples of actually manufactured jewelry. Finally, Section 6 offers conclusions.

2. Related work

CAD tools are being used more and more in artistic and aesthetic applications. In these applications the aim is not only to realize certain geometries and patterns but a main concern is the overall aesthetic result. Examples serving this purpose are presented in [8,9], where parametric sculpture generators are implemented to create and modify artifacts belonging to certain conceptual families, and to evaluate the final aesthetic result. In [10] an application concerning kinetic art, i.e., art that involves movement, is presented. A system is proposed for designing original kinetic art objects where a 3D geometric-modeling interface and a rigid-body simulation are combined. A survey on CAD methods used in another aspect of artistic expression, garment design, is provided in [11].

Jewelry design and construction is another example of combining CAD tools and aesthetics. Various commercial software packages have been developed for designing and creating CAD jewelry models such as JewelCAD [12], Rhino3D [13], ArtCAM JewelSmith/Delcam Designer [14], Matrix 3D Jewelry Design Software [15] and 3Design CAD [16]. Most of these systems provide some form of parametric and feature-based capabilities, graphical interfaces with excellent rendering capabilities, built-in libraries focused on jewelry design that include different piece settings, cut gems and stones, and advanced feature-based design tools. Some systems provide advanced functionality, such as Matrix [15], which offers the use of builders for recording design steps and for defining parameter values for parts to be used in the process. Also, the majority of these systems have the capability of exporting models to rapid prototyping machines. All of these systems provide various tools for making jewelry design a simpler and less time-consuming process.

These software packages are convenient for designing and creating various forms of generic style jewelry. However, none of these systems is appropriate for designing and creating editable CAD models of jewelry of a particular craftsmanship, such as filigree jewelry. In particular, for constructing this type of jewelry, where the fine aesthetic result is achieved through twisting, bending and combining wire strands to create complex designs, a system is required that provides the capability to: (i) create very accurate and robust solid models ready for reproduction, (ii) create parametric models of filigree jewelry that can be used for custom design and redesign and (iii) incorporate filigree designs into a solid model corresponding to a ring, bracelet, necklace, etc. In most commercial CAD systems for jewelry, designing is performed manually using various tools and usually the design steps cannot be programmed to be executed automatically and accurately. This means that each different piece of pierced jewelry would have to

Fig. 2. Silver ga'u ornamented with (a) gold, (b) silver filigree [7], and (c) detail of a golden filigree necklace [7].

be created basically from the beginning by hand which does not conform to the requirement of redesign capability. Editing is then usually achieved through history rollback, by returning to a prior design state of the model and applying modifications to the model.

Another CAD approach to designing and producing jewelry of a particular craftsmanship is ByzantineCAD [17]. ByzantineCAD is an automated, parametric CAD system for designing and producing pierced Byzantine jewelry where the user-designer sets some parameter values and ByzantineCAD creates the jewelry model that corresponds to the specified values. This provides the designer with the capacity to rapidly create custom-designed jewelry, based on the preferences of the customers. ByzantineCAD introduces a feature-based and voxel-based approach to designing jewelry, through the definition of elementary structural elements with specific attributes that are used as building blocks to construct complex pierced designs. Modern free-form jewelry can only be partially described through predefined primitives and feature libraries, due to its abstract and artistic nature which is often difficult to capture. It is usually created using curves (such as NURBS) and surface modeling techniques. An alternative approach to reconstructing free-form artifacts is presented in [18].

A parametric approach to creating carved jewelry is also presented in [19]. Voxel elements are constructed and combined to recreate jewelry depicting designs made from small carvings. This system provides both design and rapid prototyping capabilities. In [20] a parametric–feature-based jewelry modeler for designing and manufacturing Fret-worked bangles is presented. Fretwork designs are encoded as features that are recurrently removed from stock-solid bangles. Finally, Kai et al. [21] developed a reverse engineering system for re-engineering rings. This method creates basic generic models of the initial ring object. To manufacture the ring, the 3D model is transformed into a 2D representation on which the engraving of the ring design will be performed. This method is appropriate for creating blank generic models of rings that a designer will then use to create his/her own ring model.

In general, editing parts of filigree designs in commercial jewelry design systems requires in depth knowledge of feature-based design and solid modeling techniques. In contrast, our system offers an easy, semi-automated procedure for creating and customizing jewelry featuring filigree designs. In our system one defines the basic parameters that refer mostly to the appearance, size and content of the final product and then the construction of the specified model is carried out by the system. By parameterizing the process of creating filigree jewelry, it is easy to modify characteristics of the jewelry such as the size and the designs represented. Furthermore, designing a piece of filigree jewelry using a traditional CAD system may lead to models with robustness problems, which are inappropriate for manufacturing, creating therefore the need for repairing tools and techniques. The technique used in our system leads to robust models that can be directly sent to rapid prototyping machines for manufacturing without any further intervention or repair.

3. Identifying and reconstructing filigree features

Through studying the design of a series of filigree made jewelry, particularly those made by local craftsmen in northwestern Greece, we have concluded that certain primordial patterns are being used repeatedly in various combinations. Examples of filigree patterns are shown in Fig. 3. By studying the craftsmanship of filigree jewelry, a core set of patterns that are used as building blocks in a wide range of filigree designs is identified.

In this Section, we report on the characteristics of wire strands (which are the building elements of filigree jewelry) and several common elementary features observed in filigree jewelry. Finally, we report on the composition of more complex designs that are commonly used in this type of craftsmanship.

Fig. 3. Various design elements observed in filigree jewelry [7].

The following issues are considered central to achieving aesthetically acceptable filigree components:

- Derive a variety of wire strands and braids which are the building blocks of filigree jewelry.
- Combine elliptical arcs and/or circular arcs with two or more inscribed spirals of equal spacing.
- Use scaled patterns as subparts maintaining certain parameters.

Finally, in our models, wires are allowed to slightly overlap each other to enforce coherency. Wire overlapping is considered critical for manufacturability (during both 3D printing/layered manufacturing and casting) and does not affect the aesthetics of the final jewelry piece, if it remains within reasonable bounds. Experienced craftsmen and artists have determined that the edges of the resulting jewelry piece are even sharper than the original piece because of the gluing material used by the craftsmen in filigree jewelry. To have better control of the design sharpness we provide a parameter that affects the thickness of all the wires involved in feature construction by a small percentage (up to $\pm 10\%$).

3.1. Wire strands, basic shapes and features

The basic idea behind filigree craftsmanship is to twist and bend strands of wires to build complex designs. Certain properties (parameters) stand out regarding the wires used to build and decorate a filigree jewelry piece: wire strand geometry (thickness and length), the number of wire strands and the technique of combining strands to create larger decorative and more complex strands.

An *elliptic shape, an ellipsoid, a circle or a sphere* can be used either as an outline of a jewelry part or as a solid base on which the filigree design will be applied. *Archimedean* or *logarithmic spirals* are inscribed inside basic shape constructs. Rows 1 through 3 of Table 1 summarize the different types of wire strand combinations. Rows 4 through 6 illustrate combinations of elliptical arcs, circular arcs and line segments while the last rows summarize nested and inscribed patterns. For a more detailed description of all constructs the reader is referred to [22].

3.2. Braids revisited

*Braided strands*are built by sweeping a circle along a path curve (see Table 1). We have used cubic rational Bezier curves to realize braid strand path curves, a choice that offers a wider range of parameters and essentially better aesthetic results than the traditional sinusoidal braid modeling approach. Our braid approach is a set of four such curves, which are pair-wise symmetric. For cubic rational Bezier curves with control points $P_{m(t),j}$ and weights $w_{m(t),j}$ with $m(t)$ being the curve index and j

Table 1
Wire strands, basic shapes and features.

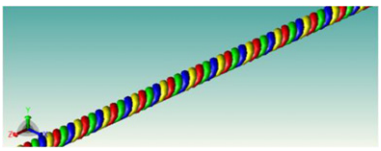
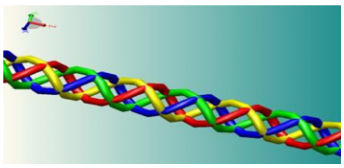
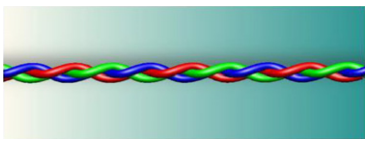
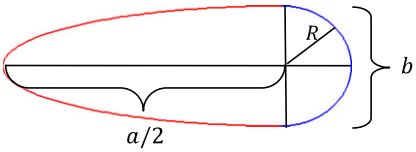
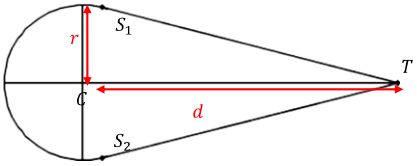
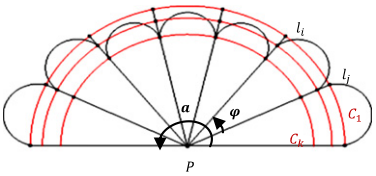
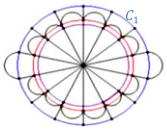
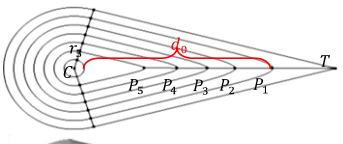
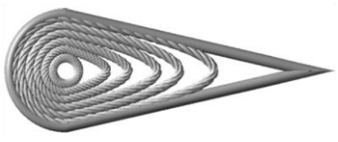
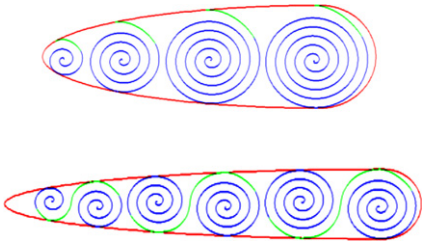
Description and construction	Parameters and constraints	Example
Twisted wire strands: are built by one or more helical strands following the same direction, derived by sweeping a circle on a helical spiral curve: $C(t) = \begin{bmatrix} r\cos(t+\theta) & r\sin(t+\theta) & ct \end{bmatrix}^{\mathrm{T}},\ t \in [0, \infty]$	r (the radius of the helix) and c (the pitch of the helix), distance d between the paths of two adjacent strands which is at least $d = c/n$, n is the number of wire strands. The radius of each strand should be slightly larger than $\frac{d}{2}$.	
Intersecting wire strands: are formed by combining two or more pairs of twisted strands. Each pair is constructed by a strand following a helical curve and a strand following the same helical curve with the same parameters but wound in the opposite direction.	Number of pairs, r (the radius of the helix), c (the pitch of the helix), distance d between the paths of two adjacent strands, which is at least $d = c/n$, n is the number of wire strands. The radius of each strand should be slightly larger than $\frac{d}{2}$.	
Braided strands are built by sweeping a circle along a path curve. Braiding can be performed on three or more strands. In filigree jewelry three strands are most often used. In the traditional sinusoidal approach each strand follows a Lissajous curve which is described by: $C(t) = \begin{bmatrix} r\cos(t+k) & r\sin 2(t+k) & ct \end{bmatrix}^{T}\ t \in [0, \infty]$	c (the pitch), r (the radius of the helix) and k which defines the displacement of the strand. By scaling and/or modifying the wire thickness of the strands, a different aesthetic result is obtained. For an analysis of the parameters and constraints see [23].	
Round Teardrop: is constructed by uniting half an ellipse and a semicircle whose centers coincide. G^1 and G^2 continuity are preserved everywhere except at the joints where only G^1 continuity is preserved. Teardrop shapes are mainly used as borders of several kinds of designs and patterns. Multiple teardrops are combined to create more complex designs.	b (the overall height), $\frac{a}{2} + R$ (the overall width). The constraint which is applied to guarantee G^1 continuity at the joints is $b = 2R$.	
Pointy Teardrop: is created by connecting a circular arc with two line segments that originate from T and are tangent to the circular arc at the endpoints S_1 and S_2, as shown in the example. S_1 and S_2 are derived from d, r and C.	distance d of the center of the circular arc from T, the radius r. G^1 continuity is preserved everywhere except at the tip of the teardrop. G^2 continuity is preserved everywhere except at the tip and at the endpoints S_1 and S_2.	
Peacock patterns: To create a peacock-like pattern, pointy teardrops are scaled and placed adjacently with a common tip point. In this work, the peacock design consists of m pointy teardrops which are sequentially placed within a center angle a of a virtual circle C_1. Given the center angle a, each pointy teardrop shape s_i is placed within a contained angle $\varphi = a/m$. To create each pointy teardrop shape, we use lines l_i and l_j dividing C_1 into the corresponding segment as edges for the teardrop and an arc created from the intersection points of C_1, l_i and l_j. To create teardrops of different lengths we assume more virtual circles C_k of radius r_k and use their intersecting points with the respective line segments.	The angles a_i formed by the line segments are then equal; however different patterns can be derived by using different configurations for each peacock "feather". : the number of teardrop segments m, the center angle a, the pattern reference point $P(x, y, z)$ which is the center of the virtual circle C_1 and the radius r_1 of C_1, which defines the actual size of the peacock and the teardrop segments. For this pattern to be robust, the following constraints must hold: Point P is common for all pointy teardrops and $\sum_{i=1}^{m} \varphi = a$.	 
Nested Teardrops: This pattern is created by defining n concentric circular arcs of radius r_i such that $r_1 > r_2 > \cdots > r_n$ and n rational cubic Bezier curves with start and end points coinciding with the start and end points of the respective circular arcs. The $(n+1) - th$ curve is defined to be a circle.	The wire strand style (simple, braided, intersecting pairs etc.) and thickness, the number of nested teardrops n, the center $C(x, y, z)$ of the n concentric arcs and the distance d_0 between the circle (C, r_{n+1}) and the tip of largest rational Bezier curve which affects the inner gap and is aesthetically important. The maximum number of teardrops n of wire strand thickness t that can be nested is $n = d_0/t$.	 

Table 1 (*continued*)

Description and construction	Parameters and constraints	Example
Teardrop shapes containing spiral designs: Each spiral should be positioned with respect to the teardrop and to the rest of the spirals. This is accomplished by defining consecutive, non-overlapping disks fitted in the teardrop. Every new spiral is drawn in one of these regions and is initially set to have the same center and a radius equal to the radius of its containing region. In the case of single spiral construction, each spiral is constructed independently. However, for spiral pairs a connector curve segment between the spirals must be established. We use the corresponding arcs of the inscribing circle for connecting the two spirals. This approach maintains G^1 continuity and provides nice aesthetic results for most configurations.	Wire thickness. Teardrop parameters: height and width of teardrop, G^1 continuity constraints for teardrop. Spiral constraints: spirals are constructed so as to be completely inside the maximal inscribed circles and to be tangent to the teardrop at two points. Spiral pairs are connected by connector Bezier patches so that the entire construct is G^1 continuous. Connector curves are required to have the curvature of the spiral at the end points so that the spiral extensions resemble to the original spiral. The spiral spacing is the same for all the spirals that are inscribed in the same teardrop. Spiral spacing is a shape parameter. The number of spirals that can fit into a teardrop is determined by the system. In the example for the top teardrop we may have up to 4 spirals and for the bottom teardrop we may have up to 6 spiral.	

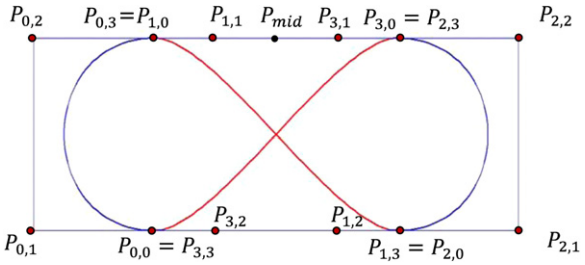


Fig. 4. Visualization of $CB(t)$, $t\epsilon[0, 4]$.

being the control point index, $m(t) = \lfloor t \rfloor \mod 4, j = 0, \ldots, 3$, the position upon the curve at any given time t is given by:

$$CB(t) = \frac{\sum_{j=0}^{3} \binom{3}{j} (t - \lfloor t \rfloor)^j (1 - (t - \lfloor t \rfloor))^{3-j} P_{m(t),j} w_{m(t),j}}{\sum_{j=0}^{3} \binom{3}{j} (t - \lfloor t \rfloor)^j (1 - (t - \lfloor t \rfloor))^{3-j} w_{m(t),j}} \quad (1)$$

$$C_i(t) = \begin{bmatrix} CB_x(t + k_i) \\ CB_y(t + k_i) \\ c(t + k_i) \end{bmatrix}, \quad t \in [0, \infty]$$

where $CB(t)$ is the XY-plane projection of the sweep path curve, and k_i is the displacement that corresponds to the i-th strand.

Fig. 4 illustrates the construction of $CB(t)$. The two pairs of symmetric curve segments that contribute to the braid are depicted in red and blue. The control points are also depicted with their indexing, revealing the direction of movement along the curve.

Two main parameters that directly specify the control points $P_{0,1}, P_{0,2}, P_{2,1}, P_{2,2}$ are the width and height of the convex hull of the braid projection. To preserve horizontal and vertical symmetry we set $w_{i,0} = w_{i,3} = 1$ and $w_{i,1} = w_{i,2}$. This reduces the number of weight parameters for each braid to two.

Control points $P_{0,3}$ and $P_{1,1}$ are then set as follows (note that $P_{0,3}$ and $P_{1,0}$ are identical and $P_{0,2}, P_{0,3}, P_{1,1}, P_{2,2}$ are collinear) based on parameters a and β: $\|P_{0,3}P_{0,2}\| = a\|P_{0,2}P_{2,2}\|$ and $\|P_{1,1}P_{0,2}\| = \beta\|P_{0,2}P_{2,2}\|$, where $a, \beta \in (0, 1/2)$. To obtain a highly symmetric aesthetic result we use $a = \frac{1}{4}$, $b = \frac{3}{8}$. The rest of the control points are computed using the horizontal and vertical symmetry of the curve segments.

Strand Placement: Braid strands should be placed so as to ensure a slight overlap of the swept circles. First, one should ensure that the braid strand centers do not collide. For an odd number n of strands we may just define $k_i = i\frac{4}{n}$, then it is easy to prove that there will be collision of the strand path curves. Indeed, the only possible scenario that leads to a collision is for a strand center point to start from the opposite symmetric location. For two locations to be anti-symmetric they need to have time–distance equal to 2. So, for two strands i, j with displacement k_i, k_j with $j > i$, the following equation must hold: $2 + i\frac{4}{k} = j\frac{4}{k} \Leftrightarrow k + 2(i - j) = 0 \Leftrightarrow k = 2(j - i)$ which implies that k is some even number.

Likewise, for an even number of strands we have to enforce non-mirror symmetry between any pair of strand centers. Fig. 5 illustrates four braid strand route-paths projected on the XY-plane.

In practice, an approach that works very well in determining the strand thickness is to derive the minimum pair-wise distance D for the set of points that consists of the strand center points of the left-side part and the symmetric center points of the right-side part (see Fig. 6). Then we choose as thickness a quantity slightly larger than D. Note that D is the maximal minimum distance between two strands since it corresponds to the tangent plane of the path curve being perpendicular to the XY-plane.

Deriving Aesthetically Optimal Braids: By comparing the curvature values of our rational Bezier approach with those of the sinusoidal approach we can adjust the parameters of the former to produce aesthetically different results.

The control points may be positioned so that the result will be as close to the shape of a sinusoid braid as possible. Then we compute the curvature at each point on the rational Bezier curves. We consider the mean of the curvature values within the specified ranges for both curves and examine only the first two curves along with the corresponding sinusoidal part, since the rest are symmetric.

Fig. 7 shows how the curvature of the rational Bezier braid progresses as the weights of the two component curves increase. The wired plane represents the mean curvature of the sinusoid braid. Indeed, the curvature of the Bezier braid rapidly becomes lower than the sinusoid braid mean curvature, whose value is $\cong$ 0.9088. Using the same positioning and weight values of $w_{0,1} = w_{0,2} = 0.8$ and $w_{1,2} = w_{1,1} = 0.7$, which are amongst the smallest, in value, weight pairs that yield curvature lower than the sinusoid braid, we acquire the curve paths shown in Fig. 7. The aesthetic improvement is apparent. Fig. 8(a) presents a 3D rendering result of a three strand braid with rational Bezier curves (red) and with the sinusoid function (green). Fig. 8(b) illustrates a smoother Bezier braid strand (red) as compared to the more curvy sinusoid braid strand (green). Note that all parameters (α, β) and weights are determined, except the number of strands that is a user defined parameter. For an odd number of strands the placement is performed automatically, while for even number of strands we have pre-placed the strands for cases up to eight strands according to the aforementioned technique.

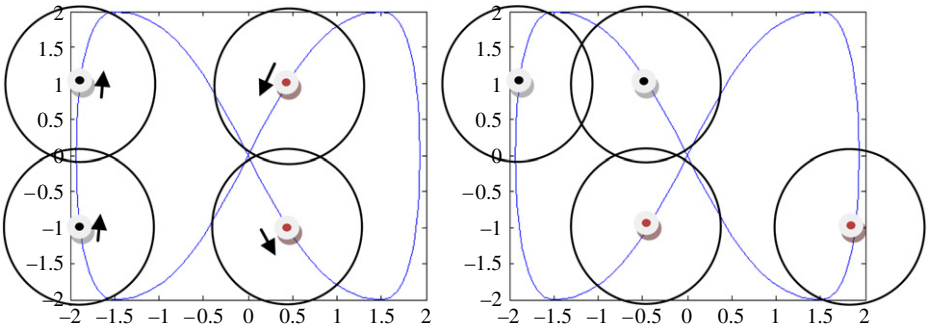


Fig. 5. Strand movement scenario.

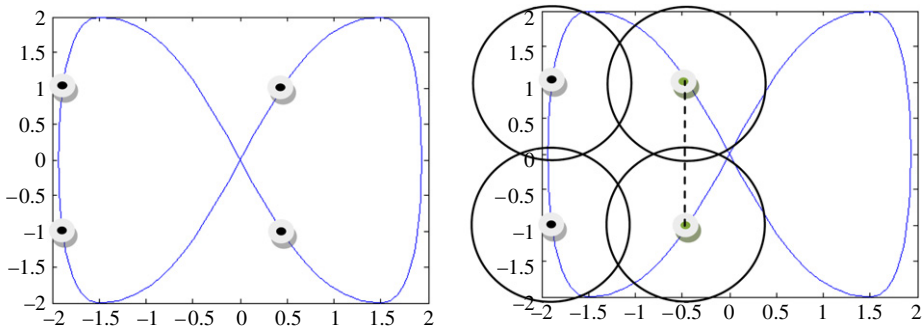


Fig. 6. Determining strand thickness for the placement on the left by using mirroring and minimum pair-wise distance picking.

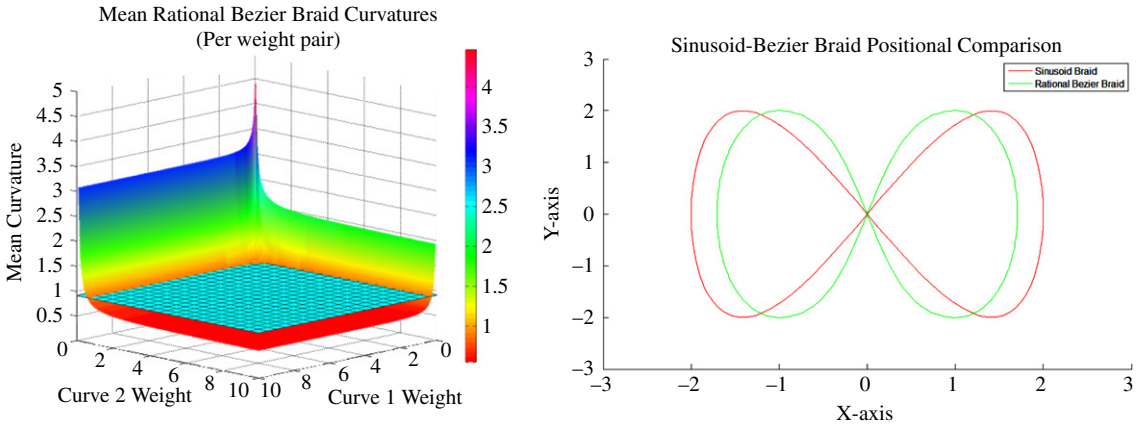


Fig. 7. Rational Bezier curve mean curvature values and the resulting optimal 2D projections.

3.3. Complex design features

Filigree craftsmanship is applied to create and decorate all types of jewelry, i.e., necklaces, pendants, rings, brooches and bracelets. There are specific complex features that are commonly used as solid base components.

Flower Design: Flower designs are commonly used in filigree jewelry. Flower designs in general are created by combining arch-shaped wires and solid spherical beads that represent the flower petals and center respectively, thus creating a daisy-like pattern.

Fig. 9 presents a 2D sketch of such a flower design. We consider concentric circles $C_1(M, R_1)$ that contain the flower design, and $C_3(M, R_2)$ corresponding to the center of the flower. Given the number of flower petals n to be used in the flower design, circles C_1 and C_3 are divided into n equal segments, whose borders are defined by lines L_i $(i = 1, \ldots, n)$. The flower petal corresponding to each segment is represented by a curve F_i $(i = 1, \ldots, n)$ that interpolates points P_{ij}, where $i = 1, \ldots, n$ and $j = 1, \ldots, k$, with k referring to the number of points used for interpolation. The start and end points used for interpolation, i.e., P_{11} and P_{15}, are located on curve C_3 and are common for adjacent flower petals, whereas the point corresponding to the tip of each petal, for example P_{13}, represents the position where C_1 is tangent to the petal.

To ensure model robustness and efficiency, we require that adjacent petals not only have common start and end points, but also have overlapping/intersecting segments. This is accomplished by considering a circular guide curve $C_2(M, R_2)$ where $R_2 = R_3 + s$ and s corresponding essentially to the length of the overlapping petal curve segment. The points where lines L_i, that divide the circles into segments, intersect with circle C_2 are used as data input for the interpolating petal curve.

Relevant parameters for the flower design: the flower center bead size, which is defined by the radius of the sphere constructed in the middle of the flower, the flower petal length, which is defined as the length $m = R_1 - R_3$, and the number of flower petals n.

Suppose the flower design is to be constructed so that it is not flat on a surface, but lies on a cone. The end user specifies an angle

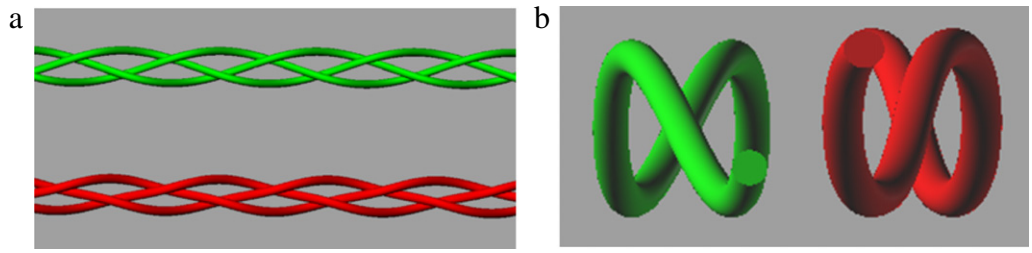


Fig. 8. (a) Rendering of braided strands created using (green) the sinusoid function and (red) rational Bezier curves. (b) Rendering of a single braid strand.

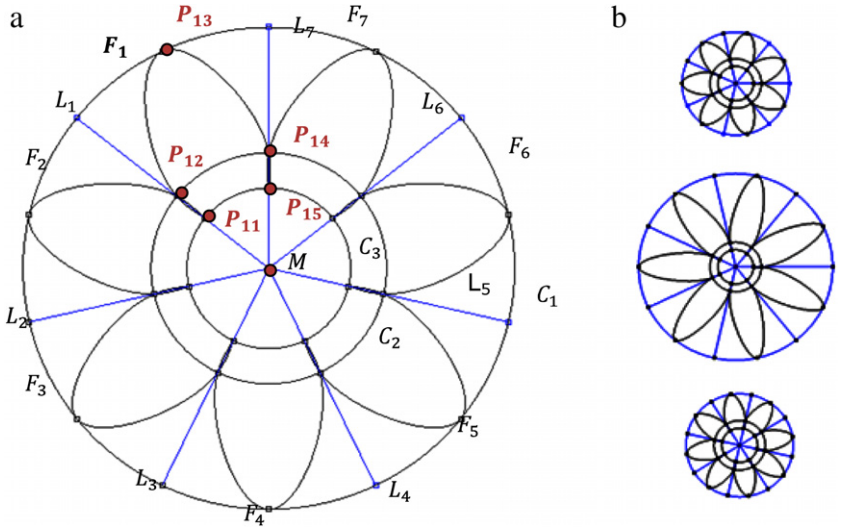


Fig. 9. (a) A 2D sketch of a flower design, (b) variations of the flower pattern.

α at which the petals are to be located and the circular pattern described above is mapped onto a conic surface so as to achieve the desired result. The final result is depicted in Fig. 10.

To create the final 3D flower pattern, wire strands are created and transformed based on the profile curves of the petals. The sphere corresponding to the center of the flower is translated along the cone axis to obtain a more aesthetically pleasing effect (Fig. 10).

Leaf Patterns. Another common pattern used in filigree jewelry is the leaf. Leaves are created by combining curves forming the shape border of the leaf with golden spirals (Fig. 11). We present two different approaches to creating leaf patterns depending on the spiral placement scheme used. The placement of the spirals in the leaf shapes is carried out by exploiting an interesting characteristic of the golden spiral. Golden spirals are logarithmic spirals where the variable b is related to the golden ratio ϕ (Fig. 12). The golden rectangle of the spiral is used to implement the placement schemes for the spirals in the leaves. Placement is performed from the bottom to the top of the leaf.

In the first placement scheme, lanceolate shaped leaf patterns are constructed. Lanceolate shapes are long and wide in the middle. The shape of the leaf is formed using two symmetric parabolas. These parabolas are expressed by the equation: $C(t) = [at^2\ 2at]^T$.

A line segment coinciding with the mirroring axis for the two parabolas is used to represent the spine of the leaf. The starting points of the decorative golden spirals used are located on this line. Spirals are placed so that each one intersects the leaf border at one point.

Initially, a golden spiral is placed such that its starting point is located on the spine of the leaf and the golden rectangle is located inside the leaf. Scaling is performed on the spiral so that it intersects the parabola at one point. The next spiral with its golden rectangle is placed on top of this spiral so they are adjacent and then this too is scaled until it intersects the parabola at one point. This process is carried out recursively either for a defined number of spirals or until no other spirals fit inside the leaf. Since the leaf is symmetric, the created spirals are mirrored onto the other half of the leaf.

Cordate shaped leaf patterns (i.e., heart-shaped leaves with the stem in the cleft) are constructed using the second placement scheme. Suppose that the bottom left corner of this golden rectangle is positioned at $P_1(0, 0)$ and its height is h_p. We seek the height of the next golden spiral that is to be placed adjacently on top such that the two spirals are tangent to each other at point $T(x_T, s)$ (Fig. 12(b)). Based on the properties of the golden rectangle the height of each newly placed golden spiral depends on the height of the previously placed golden spiral and the golden ratio. This process is performed recursively for the desired number of spirals. After placement, the border shape of the leaf is created using interpolation. It is interesting to note that the top right corners of the golden rectangles are collinear, due to the fact that the golden rectangles are similar (with a constant ratio). The start and end points that are chosen for the interpolation, control how pointy and how heart-shaped the tip and end are, respectively.

Our leaf pattern creation scheme can be generalized to apply not only to other types of spiral curves but also to more general and/or freeform curves (Fig. 12(c)). The constraint that these curves must follow is that they intersect with their corresponding bounding box at one point on the top and one point on the bottom of the bounding box and that these intersecting points are not symmetric.

Free-Form Shapes and Designs: Free-form wire shapes were not very popular in traditional filigree design. Indeed free-form design occurred usually as connectors between other patterns. Modern filigree jewelry use higher degree curves that can be derived by interactive CAGD tools (see e.g., [24,25]).

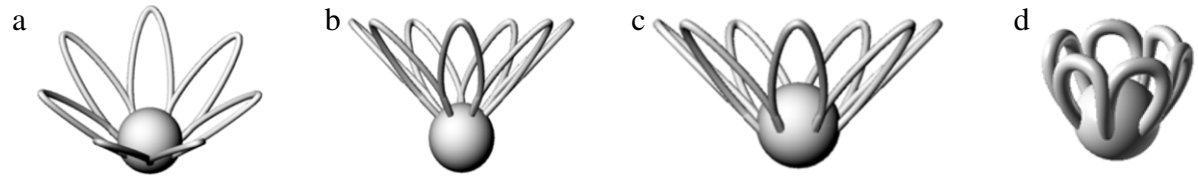


Fig. 10. (a) The final flower design, (b) initial flower pattern after construction, (c) translated beaded center for a better fit with the petals, (d) alternative design with rounded petals.

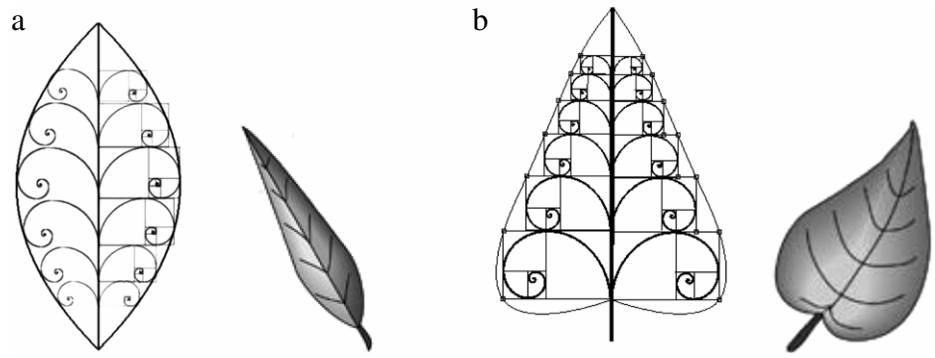


Fig. 11. (a) A pattern simulating a lanceolate shaped leaf. (b) A pattern simulating a cordate shaped leaf.

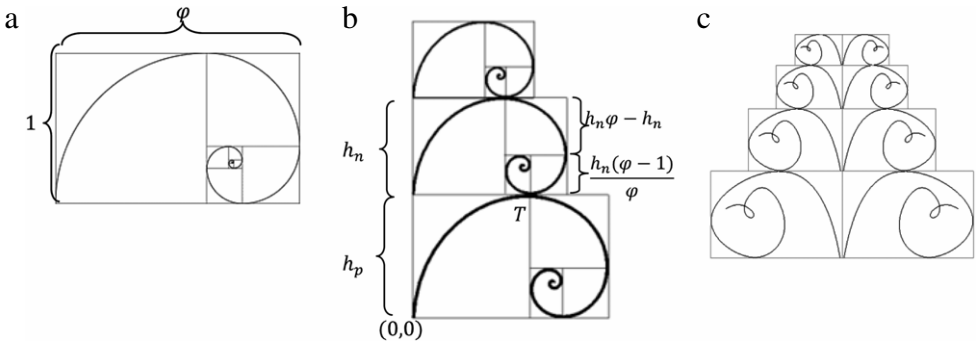


Fig. 12. (a) A golden rectangle and the corresponding golden spiral. (b) Assembling scaled tangent golden spirals to form a leaf-like pattern. (c) A leaf-like pattern using freeform curves.

3.4. Decorative elements

The final aesthetic effect of the designed jewelry is given by elements that are used for decorative purposes. *Solid beads* of different sizes are frequently used to decorate the jewelry. These beads are placed on the model and Boolean union is performed on them and their respective model component. A plane corresponding to the local surface on which the model component lies at the point where the bead is imported is used to cut the solid bead to create a flat surface. *Jewels and stones* can be embedded in/on the jewelry piece. Finally, *decorative borders* are used which are bolder, thicker, decorative strands of wires, usually in the form of braids or intersecting wire strand pairs.

4. Filigree jewelry reconstruction

An overview of the task flow of our filigree jewelry re-engineering and reconstruction tool suite is presented in Fig. 13.

A reverse engineering option (described in Section 4.1) is provided that allows the user to load a point cloud and extract information regarding feature regions and border sets that can be used as a guide during the feature reconstruction and placement process (described in Section 4.2).

For experienced users, a straightforward feature-based computer-aided design option is also available (i.e. using directly the process described in Section 4.2 on an existing jewelry template). The reverse engineering option can be used stand alone, in cases where the detail derived is sufficient to reconstruct the jewelry piece. In this case the editability of the resulting CAD model is limited.

Traditional reproducing techniques using rubber molds will only capture a rough approximation of the 3D model by effectively wrapping an elastic watertight surface around the filigree jewelry piece. Furthermore, such techniques are not capable of restoring missing parts or additive editing of the resulting mold or the corresponding wax replica that will be used for traditional investment casting.

This section makes the following key technical contributions: (i) presents carefully designed practical point cloud analysis tools for feature and symmetry detection and (ii) describes an effective user interface for creating design features and placing them with respect to each other so as to create editable CAD models of very fine detail jewelry pieces. This tool suite has been implemented using the Microsoft Visual C++ programming environment and the ACIS R18 solid modeling library provided by Spatial [26].

4.1. Jewelry re-engineering

The issue of re-engineering jewelry, and especially filigree jewelry, presents many difficulties, mainly due to the small size

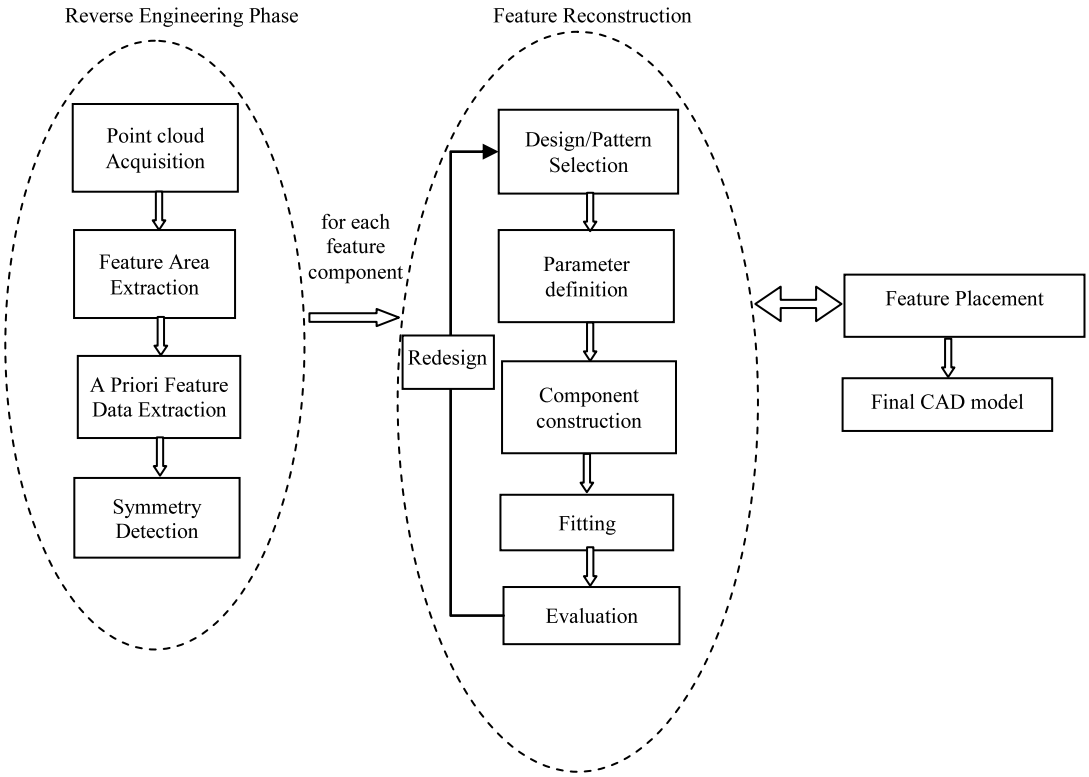


Fig. 13. An overview of our filigree jewelry reconstruction tool suite.

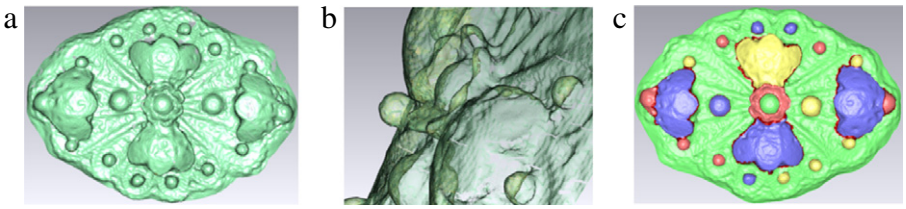


Fig. 14. (a), (b) The point cloud derived by using a 3D laser scanner on the brooch. (c) Feature component detection and extraction performed on the point cloud.

and fineness of detail. To this end, one needs to capture the features of the original jewelry model and the relationships and constraints that hold among them. We are still far from developing fully automated reverse engineering systems where there is no human intervention. It is more appropriate to design systems where the user interacts with the system and provides information that can be used to acquire a more accurate and complete CAD representation of the object. In our approach we have aimed at achieving some level of automation without sacrificing real-time response and high accuracy. This approach has been adopted successfully by general purpose reverse engineering systems [27].

A piece of filigree jewelry is usually scanned using a 3D laser scanner producing a point cloud. Due to the generally small size of jewelry and the delicacy and fineness of their decorative details, scanning techniques can adequately capture only major design features and their relative placement on the object that is being scanned. Aesthetic detail cannot be captured in a way that is useful for reconstruction. An example of a 3D point cloud of a filigree brooch, acquired using a handheld Handyscan EXASCAN laser scanner (accuracy 40 μm), is illustrated in Fig. 14. It is clear that the filigree details of the brooch cannot be inferred from the scanned data. However, we use the point cloud to extract the overall topology that will guide the identification and placement of the primitives and the overall shape of the jewelry piece.

4.1.1. Point cloud segmentation

There are various methods that can be used for the segmentation of the initial object, such as [28,29]. A thorough survey of 3D mesh segmentation methodologies examining their suitability for CAD models is presented in [30]. To detect feature regions we have adapted a method developed earlier in [31] for reverse engineering based on discovering features on the point cloud by detecting local changes in the morphology of the underlying geometry. We applied this method on meshes, where vertex adjacency information is provided a priori. By detecting rapid variations of the surface normal and measuring the concavity intensity we are able to apply a region-growing technique to extract a number of regions that represent object features (Fig. 14).

More specifically, the concavity intensity of a vertex v_i of a mesh, denoted by $I(v_i)$, is the distance of v_i from the convex hull of the mesh. This characteristic is used to detect concave features on the mesh. Then a region growing is used to detect sets of vertices that belong to individual features. Such an example is illustrated in Fig. 14(c). The feature regions that are detected correspond to the basic brooch components: four decorative ensembles, the flower centerpiece and a number of decorative beads.

By segmenting the point cloud into feature regions and boundary sets, we are able to reconstruct the boundary contours for each area using curve approximation methods, such as

[32]. Reconstructed boundaries are used in conjunction with a symmetry detection technique [33] for adapting and placing design elements.

From each detected feature the following data are retrieved:

(a) the size of the area covered by the feature by computing the corresponding bounding box,
(b) the orientation of the bounding box, based on the principal axes,
(c) the feature boundary, by reconstructing the boundary contour from the border sets extracted during mesh segmentation,
(d) the morphology of the feature area, by deriving an initial surface representation using approximation and fitting. Tensor product surfaces are used to approximate curved areas and data concerning the curvature of the area is derived, to be used later on in the design process for operations such as feature component bending.

4.1.2. Symmetry detection

We use the principal axes of the point cloud to partition it into components in which we search for symmetric feature areas. We create a proximity graph that captures local feature placement information as illustrated in Fig. 15, for each part of the object. For each feature region pair, we calculate the geodesic distances between the centroids of the corresponding regions. The graphs are then simplified by reducing the edges that correspond to large geodesic distances to facilitate region matching. This is accomplished efficiently by establishing an edge between two feature nodes if and only if the distance of their centroids is less than a distance R_M that determines the radius of the local structure that we wish to capture. In Fig. 15 this distance is set to about $\frac{1}{3}$ of the total width of the brooch.

In addition, small regions that are insignificant are attached to adjacent major regions, or if there is no adjacent major region they are merged to establish a larger region (if the result is still insignificant, the corresponding feature region is omitted). The larger feature region (that represents the platform of the object) is omitted. Finally, regions that do not belong completely in the part that we analyze are left out.

In simple cases, where meshes have an almost identical structure, matching of the corresponding graphs is trivial. For more complex cases, meshes exhibit only local structural feature similarity. Therefore, by eliminating the edges with large geodesic distances we match only local neighborhoods in the graph. These local neighborhoods still capture high-level information about the structure of the features, for example detect pattern similarities between slightly different models.

The reduced morphology graphs are used to perform a 3D alignment of the two sides of the model and establish a correspondence among feature regions as follows:

- Match the highest degree nodes in each graph and translate say the second part so that the centroids of their highest degree feature regions coincide.
- Match the send highest degree nodes and align by rotating and scaling the second part so that the centroids of their second highest degree feature regions coincide.
- Perform a 3D alignment of the two components by rotating the second part around the axis defined by the centroids of the two aligned feature regions so that the Hausdorff distance of the point cloud of the second part from the point cloud of the first part is minimized.

The remaining regions are paired according to their degree and the distance between them. Furthermore, we also take into consideration the area covered by each region by favoring the matching of regions that have similar areas. We have used the following heuristic similarity measure for matching:

$$s_{ij} = \|c_i - c_j\| \frac{\max\limits_{k\epsilon\{i,j\}} a_k}{\min\limits_{l\epsilon\{i,j\}} a_l} \frac{\max\limits_{m\epsilon\{i,j\}} d_m}{\min\limits_{n\epsilon\{i,j\}} d_n} \tag{2}$$

where c_i and c_j are the centroids of regions i and j, a_i and a_j are the corresponding areas and d_i and d_j are the degrees of the nodes in the reduced morphology graphs.

For the example shown in Fig. 14(c), feature detection derives clusters corresponding to spherical bead elements in the brooch. Using our symmetry detection method, these beads are matched in pairs and constructed to fit the respective feature areas, therefore automating significantly the process of constructing and placing the beads on the brooch. Furthermore, the detected decorative ensemble components provide data that is used for placement of the ensembles during reconstruction, i.e., ensemble orientation and exact location in reference to the base component. Additional approximate measures are acquired from the detected features in reference to component size and bending angles by exploiting the oriented bounding box of each feature component data point set. By constructing and placing one of the ensembles and using transformations to modify and place the remaining, we ensure robustness and fairness in the model. Also, point sets are extracted that correspond to feature region boundaries from which free-form contours are reconstructed to approximate the border, providing guide contours for aligning and fitting the design component elements. Finally, the base component of the brooch is an area on which surface approximation can be performed to derive a basic shape to be used in the feature-based design process.

The user interface for reverse engineering in ReJCAD is displayed in Fig. 16. The user imports the point cloud that she/he wants to process and performs feature detection to produce an initial list of all the feature regions and borders. The user can process each feature region detected and either merge it with others or split it into more components. Symmetry detection is carried out to produce a suggested list of symmetries which the user can either confirm or reject. Symmetries are suggested to the end user in pairs of features, while an option is always provided to the user to add his/her own symmetries between pairs of features.

4.2. Feature reconstruction and placement

The feature-based design process is carried out for each feature area that is detected and extracted from the reverse engineering phase or for each feature component that the user plans on creating based on her/his design concept.

Initially the user creates a new feature by picking a pattern/design and defining the parameter values for the corresponding attributes. If re-engineering has been performed prior to the design process, then the user can choose a feature region from the feature region list as a guide for initial parameter values and placement. Intra-feature constraints of designs and patterns are enforced during feature definition so as to make the corresponding feature rigid. After construction the feature is uploaded into a feature list that contains all the feature components (every instance of a feature) that are part of the final jewelry model. A feature component is redesigned by editing its parameters, reconstructing it and updating the feature list. The user can also create profile curves by curve approximation on border point sets obtained through reverse engineering. These profile curves are used either as guides for placement of a feature or for bending other components along these curves. The user has the ability to transform the component by translation, rotation, scaling, or bending.

After each feature is constructed it is considered as a rigid body. Feature placement is realized through (i) defining inter-feature

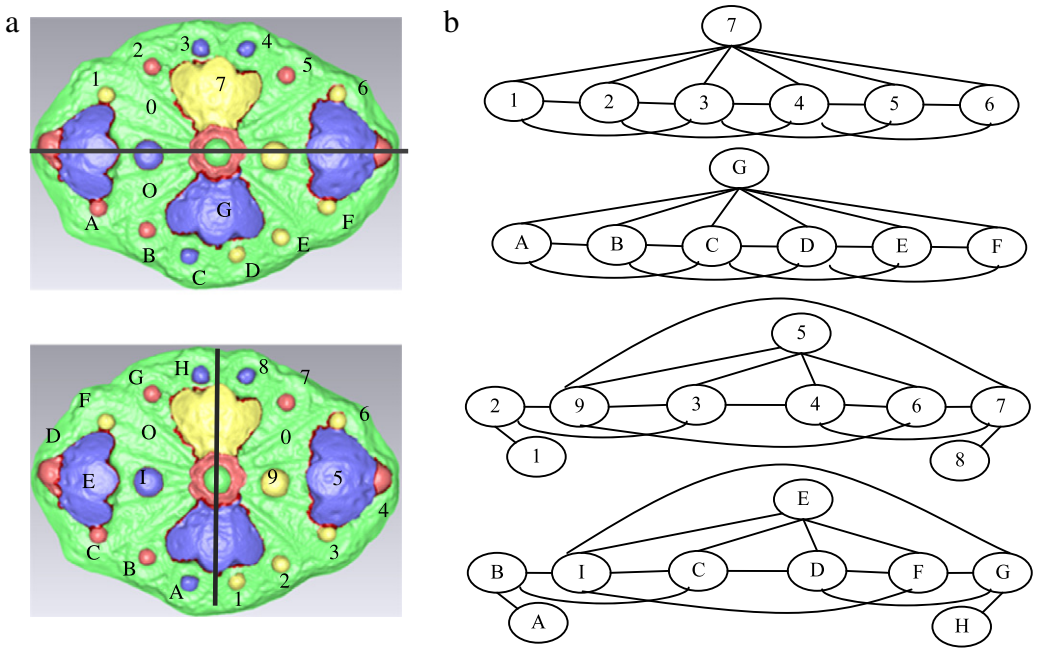


Fig. 15. (a) The original model divided into two components, according to the major and minor principal axes. (b) The corresponding morphology graphs for each component.

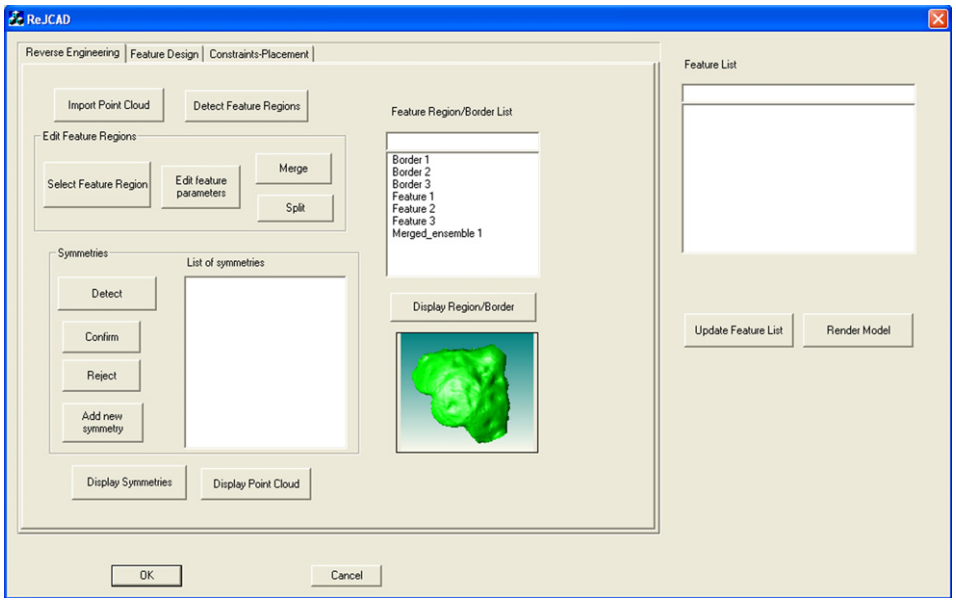


Fig. 16. A snapshot of the ReJCAD user interface.

constraints that determine the relative feature placement (with respect to each other) and (ii) fixing feature geometry. Constraints are derived and enforced on the reconstructed jewelry representation to maintain model robustness (manufacturability), conformity to filigree craftsmanship (e.g. concentric or aligned patterns) and to support editability through constraints. Constraints related to symmetry are derived through re-engineering and aim at enforcing symmetry, whereas other inter-feature constraints are user defined. The inter-feature constraints that are supported in ReJCAD mainly refer to component size, bending and placement. Specifically, size constraints are set either as equalities or symbolic relations. This also holds for bending angles of feature components. Placement constraints refer to coincidence, tangency and distance characteristics. More specifically we support the following categories of inter-feature constraints: Determining distances, angles, fixing of geometric elements, symbolic relations of angles, distances and other geometric characteristics such as radius, area, and volume (symmetry is enforced through relations), "on" constraints, tangency (supported only for a limited repertoire of geometric primitives), and inequalities (that should be used with caution, since they can be enforced only during the second step of nonlinear optimization).

We use a hybrid geometric constraint solving system. First building on the principles of [34–36] we detect minimal concurrent non-linear systems and eliminate symbolic relations. Then we employ interior point optimization to solve each linear system of geometric constraints [37]. Interior point optimization is a fast local optimization technique that can be combined with overconstraining to find efficiently unique solutions for the minimal system of geometric constraints. Root multiplicity is

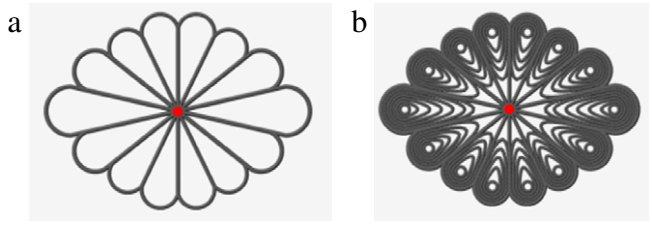


Fig. 17. (a) A wire outlined peacock pattern as a brooch base component. (b) Brooch base component featuring nested teardrop patterns.

enabled by combining the subsystems to derive a solution for the entire system of geometric constraints using the algorithms described in [34].

After the end user is done placing and editing the new component, a test can be performed on the feature to determine if the final placement is satisfactory for a Boolean union to be performed to integrate the feature into the final model. The actual union can be performed at a later phase, when the user has finished with the entire model and its re-design. The final model is checked for robustness and plausible problems that cannot be handled automatically are reported to the user for editing. More specifically disjoint parts are reported to the user, non-manifold edges and faces are corrected automatically, almost non-manifold faces and edges are reported to the user, dangling edges and faces are automatically eliminated and holes are automatically covered. The robust model is exported as an STL file that can be submitted directly for reproduction.

5. ReJCAD evaluation

In this section we provide a case study of reconstructing a filigree brooch, present a usability study that has been performed to evaluate our application and discuss the robustness and manufacturability of the reconstructed CAD models.

5.1. A case study

We present an example of designing and creating a filigree brooch similar to the one illustrated in Fig. 1. This piece of jewelry is made up of essentially two levels of components, the base or main component and the decorative and complementary elements/subcomponents placed on top.

Initially, the base component of the brooch is created. From the reverse engineering process we cannot obtain sufficient data about the patterns used for this component, however the general shape, morphology and size of the area are derived. This component is a complex pattern made up of pointy teardrops placed in a peacock pattern. The peacock design covers an angle of 360° and consists of 14 pointy teardrop segments. Three circle guide curves are defined to create three different sizes of teardrops (Fig. 17). A simple wire strand type is chosen to create the pattern outline. The origin/center O of the brooch is placed at the center of the peacock.

Next, the design to be featured in each segment is chosen. The same design pattern is chosen for each segment, the nested teardrop design. The user specifies style and thickness of the wire strands and the number of strands (Fig. 17). The base component is decorated with solid beads and a decorative braided border. The decorative border is placed by sampling points on the outline of the base component or using the shape boundary reconstructed from re-engineering, performing curve fitting to derive the profile curve and then bending a straight braided complex wire of appropriate length to follow this profile curve (Fig. 19(a)). In this case study, the border contour on which the decorative braid (Fig. 20) is fitted on is created using a segment of the boundary contour of each pointy teardrop and connecting these segments with quadratic rational Bezier curves with G^1 continuity. A parameter k is defined which expresses the percentage of the boundary contour of the pointy teardrop that is used as part of the braid contour. A larger k means that a larger segment of the available pointy teardrop boundary is used as part of the braid profile contour; therefore the braid covers more of the surface of the pointy teardrop. Quadratic rational Bezier curves are used to connect adjacent curve segments. By modifying the weight of the middle control point, we adjust the depth of the braid contour, which aesthetically affects how closely the braid contour follows the base component.

We create additional model components that will be integrated into the base component to improve its aesthetics. The basic shape that is chosen is the pointy teardrop that is combined into 3-piece ensembles. The user-designer specifies the size and the location of the subcomponent in reference to the base component. The subcomponents are placed parallel on top of the base component with the tips of the 3-piece subcomponents coinciding with the positions specified by the user. Bending is performed on each 3-piece ensemble around the bending axis (Fig. 18) and toward the bending direction specified (concave or convex result). The bending angle is specified by the end-user or is extracted from re-engineering based on the curvature of the corresponding feature region. After bending, each component is translated so that each end of the bent component intersects with the base component.

To make the 3-piece peacock ensembles placed on the ends of the brooch more robust from a manufacturing point of view, and aesthetically pleasing, we place solid beads at the tips. The brooch is decorated with a flower pattern placed at its center (Fig. 19). The center of the flower is also used as the decorative solid bead end of the middle peacock ensembles.

The final step in the brooch reconstruction process is bending the brooch model in two directions as shown in Fig. 19. Bending is performed symmetrically along each axis. In general, during feature reconstruction and placement, several constraints are imposed on the model of the filigree brooch that enforce symmetry and relative positioning of features (see Figs. 19 and 20).

Custom design and redesign is realized by selecting feature components and modifying their parameter values. For example, the filigree brooch created above can be modified to achieve the result shown in Fig. 21. The decorative flower center has been removed and the two symmetric peacock ensembles on the ends

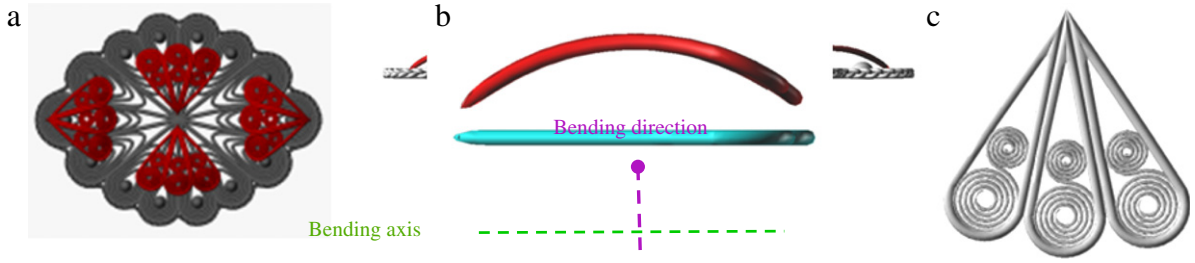


Fig. 18. (a), (b) A front and side view of the brooch decorated with peacock-like ensembles. (c) The 3-piece peacock ensemble.

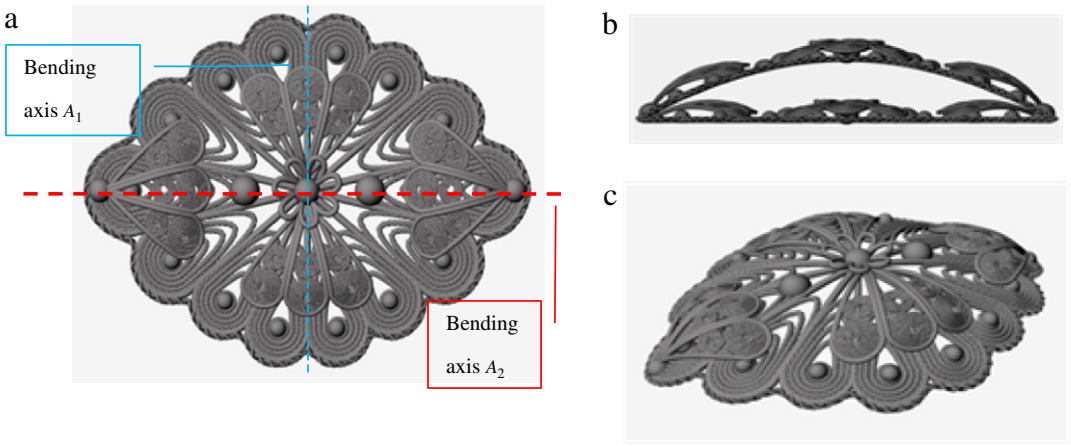


Fig. 19. (a) A 3D model of a filigree brooch prior to bending, (b) side views of the model before and after bending in one direction and (c) The 3D CAD model of the brooch after bending is performed in two directions.

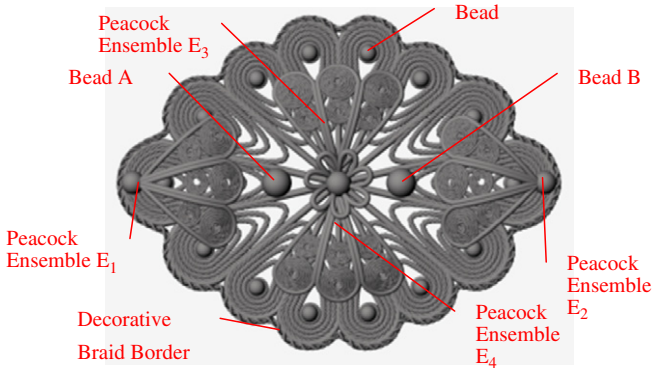


Fig. 20. Elements of the brooch on which constraints are enforced.

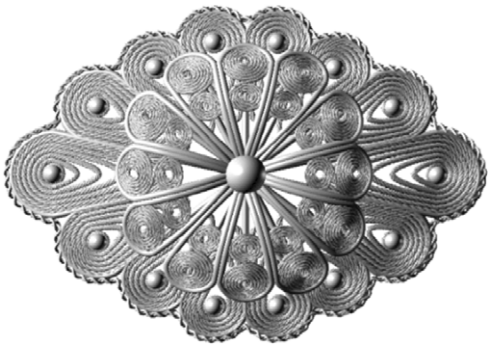


Fig. 21. A modified version of the filigree brooch.

of the brooch have been translated to the brooch center, such that their tips touch. Also all the peacock ensembles are slightly scaled up, to a point where the tips of the top and bottom ensembles touch the decorative beads of the base component.

5.2. Usability evaluation

We have evaluated our application by conducting a training process. Twelve experienced CAD users (three engineers with experience on Rhino, four graduate students, two jewelry designers and three postdocs all not involved in the project research and development) underwent a training session on how to use the application, created a filigree jewelry model under the guidance of the developers and then were asked to repeat the procedure by themselves by creating a simplified filigree brooch. Following this, they were asked to fill in a questionnaire evaluating the application by grading specific aspects of it on a scale of 0 (weak) to 7 (strong). The questions regarding the application were questions that evaluated two of the major principles of usability: Learnability (Predictability, Familiarity, Generalizability, Consistency) and Robustness (Observability, Recoverability, Task Conformance) and the three major parameters of usability: Satisfaction, Efficiency and Effectiveness.

The results derived an overall average of: 5.2 out of 7 for efficiency, 5.8 out of 7 for effectiveness and 4.4 out of 7 for subjective satisfaction. The evaluation revealed that users were able to create and redesign a 3D CAD model of filigree jewelry with satisfactory precision. The general notion among the users was that the tools provided by the application were sufficient for the task at hand and a precise model can be constructed for reproduction, however an advanced level of knowledge and experience in CAD systems is required to be able to use the application effectively and efficiently.

5.3. Robustness and manufacturability

The CAD model of ReJCAD is exported in STL format and is robust and ready for manufacturing. Robustness is ensured by performing a number of validity tests on the resulting STL model that results in removing dangling faces, filling holes and correcting various inconsistencies. Important for the manufacturability is the concept of feature element overlapping which guarantees stability and direct casting manufacturability. To this end, besides the per feature wire thickness parameter we provide a global wire thickness parameter that globally alters wire diameter by $\pm 10\%$ in all features. When decreasing this parameter the object becomes sharper and when increasing this parameter the object becomes more coherent. Subsequently, a number of manufacturing methods can be used to produce jewelry or ornamental pieces.

For reconstructing jewelry pieces of high quality from a robust STL description, direct investment casting of precious metals is the process that yields fine jewelry pieces: First we use rapid prototyping with high precision machines (with accuracy in the area of 10 to 70 μm voxels) to produce a high quality resin (or other material) prototype by using layered manufacturing usually by high resolution photopolymerism technology per layer (stereolithography). Then for direct investment casting, an investment is created – that is a negative mold of the jewelry piece usually made of gypsum – from the resin that is subsequently used for precious metal casting. Note that the resin is such that it vanished completely without leaving residue and without

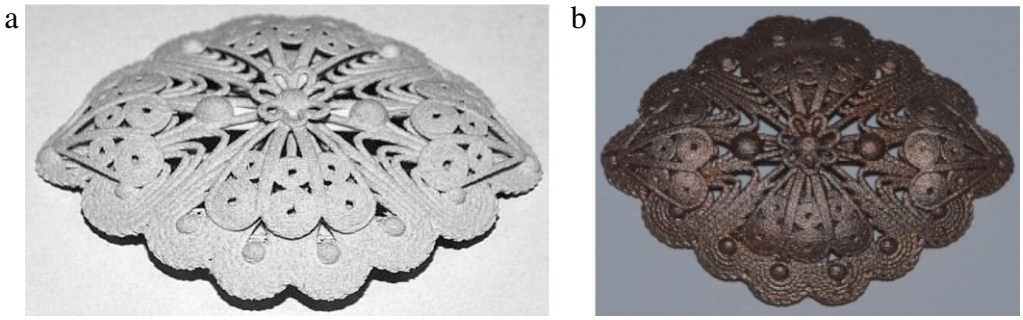


Fig. 22. (a) An initial synthetic prototype manufactured by a 3D printer and (b) the corresponding unfinished reconstructed jewelry piece after being processed with strengthening infiltration and metal coating.

interacting with the investment usually at moderate burn out temperatures. We have checked the feasibility of this process on a Perfactory Mini Multi Lens by EnvisionTec with SXGA+ resolution, 75 mm lens, and ERM (Enhanced Resolution Module) XY voxel size 21 μm and dynamic voxel thickness 15 μm (depending on the material) with the PIC100 resin for direct investment casting (see also [38]).

For an overview of such technologies the reader is referred to [39]. Several state of the art machines supporting these production processes can be found in [40,38].

For creating more inexpensive faux bijoux pieces, current techniques use ceramic or other highly resistant synthetic material for 3D printing and then a nickel, chrome of copper based mixture of metal coating is applied. The result is solid and wearable and it may become colorful by using color agents in the coating material or subsequent silver or gold plating to improve appearance and wearability. Metal coating results in deterioration of resolution but it is much more affordable for producing a small number of pieces.

A prototype of the CAD model is shown in Fig. 22(a), which was produced using a Z450 (Zcorp® 3D Printer) and made out of zp130 plaster powder containing crystalline silica and vinyl polymer. Fig. 22(b) shows a final reconstructed jewelry piece after being processed with medium strength infiltrant and metal coated. The size of both pieces is approximately 2 in by 1.2 in which is slightly larger than the original brooch size (0.9 in. by 1.1 in.). Note that some detail has vanished as a result of the metal coating process.

6. Conclusions

Filigree jewelry requires highly detailed and complex craftsmanship. Until now, there has been no technique to reconstruct and manufacture such pieces through CAD/CAM models. ReJCAD provides the means for performing filigree jewelry reconstruction through an effective partially automated process that takes a point cloud of the object as input and extracts boundary curves and other information referring to placement and symmetry. This data is then used to guide reconstruction supported by an extensive filigree feature library. We have proposed schemes for representing and modeling design features that are commonly used in filigree craftsmanship. Overall, we have reported on the development of a versatile and robust system for reconstructing filigree jewelry and we have demonstrated its usefulness through an example case, a usability evaluation and examples of actually manufactured jewelry.

The final result has been reviewed by jewelry artists and local craftsmen. They have all found the result of very fine detail especially the part of engraving which usually suffers through reconstruction. The robust final outcome can be manufactured through accurate resin prototyping after making a highly detailed gypsum investment. Some experimentation should be performed based on the complexity of the investment and the metal mixture used for achieving best results for castings using precious metals.

The overall CAD model or single features can be easily imported and used by other jewelry CAD software such as Rhino or Matrix by exporting the models through ReJCAD to SAT, 3DM or other appropriate file format for adding precious stones or performing other artistic alterations and corrections in the expense of losing part of the constraint-based editing capabilities.

References

[1] Malagoli MG. CAD/CAM technology — transforming the Goldsmith's workshop. Gold Technology 2002;34:31–5.
[2] Molinari LC, Megazzini MC. The role of CAD/CAM in the modern jewellery business. Gold Technology 1998;23:3–7.
[3] Hoffmann CM, Joan-Arinyo R. On user-defined features. Computer Aided Design 1998;30(5):321–32.
[4] Bronsvoort W, Bidarra R, van der Meiden H, Tutenel T. The increasing role of semantics in object modeling. Computer Aided Design and Applications 2010; 7(3):431–40.
[5] Bohm W. Design opportunities through production technology. Gold Technology 1998;23:8–11.
[6] Andrews C. Ancient egyptian jewellery. British Museum Publications; 1990.
[7] Untracht O. Jewelry of India. Thames and Hudson; 1997.
[8] Sequin C. CAD tools for aesthetic engineering. Computer-Aided Design 2005; 37:737–50.
[9] Sequin C. Computer-aided design and realization of geometrical sculptures. Computer-Aided Designs and Applications 2007;4(5):671–81.
[10] Furuta Y, Mitani J, Igarashi T, Fukui Y. Kinetic art design system comprising rigid body simulation. Computer Aided Design and Applications 2010;7(4): 533–46.
[11] Liu Y, Zhang D, Yuen M. A survey on CAD methods in 3D garment design. Computers in Industry 2010;61:576–93.
[12] Jewellery CAD/CAM Limited. JewelCAD. Available from: http://www.jcadcam.com.
[13] Robert McNeel & Associates. Rhinoceros. NURBS modleing for Windows. Available from: http://www.rhino3d.com/jewelry.htm.
[14] Delcam. ArtCAM JewelSmith. Available from: http://www.artcamjewelsmith.com/.
[15] Gemvision. Matrix 3D jewelry design software. Available from: http://www.gemvision.com/html/products/matrix/matrix.html.
[16] 3Design. 3Design CAD. Available from: http://www.3design.com/.
[17] Stamati V, Fudos I. A parametric feature-based CAD system for reproducing traditional pierced jewellery. Computer Aided Design 2005;3(74):431.
[18] Stamati V, Fudos I. Building editable Brep models from unorganized point clouds. Department of Computer Science, University of Ioannina Greece; 2010. TR.2010-04.
[19] Gulati V, Singh H, Tandon P. A parametric voxel based unified modeler for creating carved jewerly. Computer Aided Design and Applications 2008;5(6): 811–21.
[20] Gulati V, Tandon P, Singh H. A jewelry modeler for the fret-worked bangles. International Journal of Computer Applications 2010;2(2):76–80.
[21] Kai CC, Gay R. CAD/CAM/CAE for ring design and manufacture. Computer-Aided Engineering 1991;(February):13–24.
[22] Stamati V, Antonopoulos G, Azariadis Ph, Fudos I. ReJCAD: a feature-based CAD system for reconstructing traditional filigree jewelry. Greece: Department of Computer Science, University of Ioannina; 2011. TR-2011-4.
[23] Bates L, Wyvill B, Yao C. The visualization of knots and braids. The Journal of Visualization and Computer Animation 1992;3(2):91–104.
[24] Llamas I, Kim B, Gargus J, Rossignac J, Shaw C. Twister: a space-warp operator for the two-handed editing of 3D shapes. In: ACM SIGGRAPH 2003. ACM; 2003.

[25] Rossignac J, Schaefer S. JSplines. Journal of Computer-Aided Design 2008; 40(10–11):1024–32.
[26] Spatial. ACIS 3D modeling. Available from: http://www.spatial.com.
[27] Fisher RB. Applying knowledge to reverse engineering problems. Computer-Aided Design 2004;36:501–10.
[28] Varady T, Facello M, Terek Z. Automatic extraction of surface structures in digital shape reconstruction. Computer Aided Design 2007;39: 379–388.
[29] Stamati V. Reconstructing feature-based CAD models based on point cloud morphology. Ioannina: Dept. of Computer Science, University of Ioannina; 2008.
[30] Agathos A, Pratikakis I, Perantonis S, Sapidis N, Azariadis P. 3D mesh segmentation methodologies for CAD applications. Computer-Aided Design and Applications 2007;4(6):827–42.
[31] Stamati V, Fudos I. A feature-based approach to re-engineering objects of freeform design by exploiting point cloud morphology. In: SPM 2007: ACM symposium on solid and physical modeling. 2007.
[32] Stamati V, Fudos I. On reconstructing 3D feature boundaries. Computer Aided Design and Applications 2008;5(1–4):316–24.
[33] Athanasiadis T, Fudos I, Nikou C, Stamati V. Feature-based 3D morphing based on geometrically constrained sphere mapping optimization. In: SAC 2010. 2010.
[34] Fudos I, Hoffmann CM. A graph-constructive approach to solving systems of geometric constraints. ACM Transactions on Graphics 1997;16(2):179–216.
[35] Hoffmann CM, Joan-Arinyo R. Symbolic constraints in constructive geometric constraint solving. Journal of Symbolic Computation 1997;23:287–99.
[36] de Regt R, van der Meiden H, Bronsvoort W. A workbench for geometric constraint solving. Computer Aided Design and Applications 2008;5(1–4): 471–82.
[37] Wachter A, Biegler LT. On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. Mathematical Programming 2006;106(1):25–57.
[38] EnvisionTec. Computer aided modeling devices. Available from: http://www.envisiontec.de.
[39] Wannarumon S, Bohez ELJ. Rapid prototyping and tooling technology in jewelry CAD. Computer Aided Design and Applications 2004;1(1–4):569–76.
[40] LB Jewelry Design, Inc. Jewelry design. Available from: http://www.lbjewelrydesign.com/.