STAG: Smart Tools and Applications in Graphics (2019)
M. Agus, M. Corsini and R. Pintus (Editors)

HT-based recognition of patterns on 3D shapes using a dictionary of mathematical curves

C. Romanengo[1], S. Biasotti[1], and B. Falcidieno[1]

[1]Istituto di Matematica Applicata e Tecnologie Informatiche 'E. Magenes' - CNR, Italy

Abstract

Characteristic curves play a fundamental role in the way a shape is perceived and illustrated. To address the curve recognition problem on surfaces, we adopt a generalisation of the Hough Transform (HT) which is able to deal with mathematical curves. In particular, we extend the set of curves so far adopted for curve recognition with the HT and propose a new dictionary of curves to be selected as templates. In addition, we introduce rules of composition and aggregation of curves into patterns, not limiting the recognition to a single curve at a time. Our method recognises various curves and patterns, possibly compound on a 3D surface. It selects the most suitable profile in a family of curves and, deriving from the HT, it is robust to noise and able to deal with data incompleteness. The system we have implemented is open and allows new additions of curves in the dictionary of functions already available.

CCS Concepts

• ***Computing methodologies*** → ***Shape modeling;*** *Representation of mathematical objects;*

1. Introduction

The characterisation and recognition of curve patterns on surfaces is a well known problem in Computer Graphics. Characteristic curves, that is curves characterising a shape feature, are useful for visual shape illustration [KST08] and perception studies support these curves as an effective choice for representing the salient parts of a 3D model [CGL*08, HT11].

Given a set of (characteristic or feature) points, there is a large literature for the curve fitting problem with spline-based curves, [Far93, Shi95, PT97, HPW05, DIOHS08, APM15]. Being based on a local curve interpolation, such a class of methods is not able to recognise entire curves, to complete missing parts, and to assess if a pattern is repeated at different scales. Similarly, also methods based on co-occurrence analysis approaches [SJW*11, LWWS15] fit characteristics curves in terms of poly-lines (i.e. connected sequences of segments) that do not have any global equation.

Alternately, to recognise a global approximation and infer specific curve parameters it is possible to approximate characteristic curves with some specific family of curves. For instance, the *natural 3D spiral* [HT11] and the *3D Euler spiral* [HT12] have been proposed as a natural way to describe line drawings and silhouettes showing their suitability for shape completion and repair. More recently, the extension of the HT formulation to curves with an explicit algebraic form [BR12, BMP13] has permitted the development of methods for image curve recognition to surfaces [TBF18].

In this paper we focus on recognising shape feature curves on 3D shapes, represented by sets of points and approximated with mathematical curves via the Hough transform. This paper extends the dictionary of curves for the HT on surfaces introduced in [TBF18] and first used in [MGM*19] to recognise and characterise feature curves within a benchmark of surfaces. The focus here is the description of a rich family of primitive curves, how these curves can be adopted to characterise a pattern and the flexibility of the method to meet different shapes. These curves can be combined to obtain more complex curve configurations and further extended, for instance taking advantage of a curve dictionary [Shi95]. In particular, our method recognises various curves and patterns, possibly compound, and selects the most suitable profile in a family of curves. Deriving from the HT, the method inherits the robustness to noise and the capability of dealing with data incompleteness as for the degraded and broken 3D artefacts.

The reminder of the paper is organised as follows. Sections 2 and 3 briefly survey the HT literature and the mathematical formulation adopted in our method. The dictionary used in our system is detailed in the Section 4, where the main curve properties are described and a detailed explanation on how to pass from a mathematical curve formulation to an HT compliant formulation is provided. An algorithmic description of our procedure and some experimental results are provided in Section 5. Conclusions and final remarks end the paper.

DOI: 10.2312/stag.20191361

2. Previous work

The Hough Transform answers to both the need of robustness to noise and data incompleteness and the flexibility of the template curve. Since its original definition, [Hou62] for straight line detection, the HT has been extensively used for the recognition of circles and ellipses in [DH72] and then generalised to the identification of non-analytic profiles in images [Bal81]. This generalisation of the HT in [Bal81] is able to detect an arbitrary (but fixed) shape using a look-up table to drive the template matching, still retaining the HT robustness. Nevertheless, this HT generalisation adopts a brute force approach that considers all the possible orientations and scales of the input shape, thus the number of parameters in its process is considerably high and prevent its use in the 3D space. Further, being based on a single-shape template it cannot adequately handle similar shapes, as in the case of different instances of the same shape, which are comparable but not identical, e.g. petals and leaves. For more details on the HT and its extensions we refer to [MC15, KTT99].

More recently, theoretical foundations have been laid to extend the HT technique to the detection of algebraic objects of codimension greater than one (for instance algebraic space curves) taking advantage of various families of algebraic planar curves (see [BR12] and [BMP13]). Thanks to these theoretical foundations, the HT techniques have been able to include special classes of curves whose algebraic forms are known, but significantly more complex than straight lines or conics. Specifically, irreducible algebraic plane curves like elliptic curves, curves with 3 convexities, Wassenaar curves, conchoids of Slüse and piriform curves have been considered and the standard line and conic detection algorithm has been extended to detect these curves [MPCB15a]. In addition, also collections of different algebraic pieces from the same family or low-degree piece-wise polynomial curves have been introduced in [RBM15, CRS18].

In 3D, other variants of HT have been introduced and used, but as far as we know none of them exploits the huge variety of possible curves. For instance, in [OLA14] the HT has been employed to identify recurring straight line elements on the walls of buildings. In that application, the HT is applied only to planar point sets and line elements are clustered according to their angle with respect to a main wall direction; in this sense, the Hough aggregator is used to select the directions of the feature lines (horizontal, vertical, slanting) one at a time. More recently, [TBF18] have proposed an implementation of the theory described in [BR12] for curves in the space. In this work, also non-algebraic curves have been included in the HT paradigm. Taking advantage of the assumption that characteristic curves can be locally projected onto a fitting plane, the HT is evaluated for a planar-approximation of the curve and then re-projected on the object surface. The method showed its effectiveness for the recognition of features, in particular anatomical features like eyes and mouths. In this paper, starting from [TBF18], we provide and discuss a larger dictionary of curves and, besides the curve composition as a product of families of curves, we also present how repetition rules (like rotation, translation, etc.) can be applied to effectively characterise and recognise complex curve patterns.

3. Preliminary concepts on the Hough transform

In this Section, we briefly summarise the theoretical background at the basis of our approach. Let $\mathbb{K}$ be a field ($\mathbb{K}$ can be either $\mathbb{R}$ or $\mathbb{C}$). Then, we denote $\mathbb{A}^n_x(\mathbb{K})$ and $\mathbb{A}^t_\Lambda(\mathbb{K})$, respectively, the n-dimensional and the t-dimensional affine space over $\mathbb{K}$, with $x = (x_1, ..., x_n)$ and the indeterminate $\Lambda = (\Lambda_1, \ldots, \Lambda_t)$. Since a curve $\mathcal{C}$ is defined as the zero locus of a function f, a parameter dependent family of curves can be described by functions f_λ as:

$$\mathcal{F} = \{\mathcal{C}_\lambda : f_\lambda(x) = 0 \mid \lambda \in \mathcal{U}\},$$

where $\mathcal{U}$ is an open set of the parameter space $\mathbb{A}^t_\Lambda(\mathbb{K})$. Then, given a point $P \in \mathbb{A}^n_x(\mathbb{K})$, the HT of the point P with respect to the family $\mathcal{F}$ is $\Gamma_P(\mathcal{F}) = \{f_\Lambda(P) = 0\}$ in $\mathbb{A}^t_\Lambda(\mathbb{K})$. The classical HT in the affine plane $\mathbb{A}^2_{(x,y)}(\mathbb{K})$ is based on the *point-line duality* as follows: each point $P = (x_P, y_P)$ of a straight line $l_{a,b} : y = ax + b$ $(a, b \in \mathbb{K})$ corresponds to a line $\Gamma_P : y_P = Ax_P + B$ lying in the parameter plane $\mathbb{A}^2_{(A,B)}(\mathbb{K})$. Furthermore, the straight line l corresponds to a single point in the parameter plane, which is the intersection of all lines Γ_P, varying $P \in l$.

The extension in [BR12] expressed the HT in an implicit formulation:

$$P \in \mathcal{C}_\lambda \iff f_\lambda(x_P, y_P) = 0 \iff \lambda \in \Gamma_P(\mathcal{F}).$$

Such a definition provides the criterion for the automatic selection of the parameters because $\lambda \in \bigcap_{P \in \mathcal{C}_\lambda} \Gamma_P(\mathcal{F})$. In particular, it highlights that this formulation is able to deal with classes of curves larger than the traditional ones (i.e., lines, circles, ellipses) such as the polynomial curves.

4. From a mathematical curve expression to an HT-compliant one

In this Section, we describe the library of families of mathematical curves we have developed in our system for HT-driven curve recognition. According to [BMP13], polynomial curves are suitable for the implicit HT formulation described in Section 3, but the method is not limited to them. Indeed, in [TBF18] a first set of families of mathematical curves suitable for recognising features on surfaces was already proposed, namely the *citrus curve*, the *Archimedean spiral*, the *Lamet curve*, the *m-convexities curve*, the *geometric petal curve* of type *(A)*, see examples in Figure 1.

In this paper we extend the dictionary to six additional families of curves: the *geometric petal curve* of type *(B)*, the *elliptic curve*, the *lemniscate of Bernoulli*, the *egg of Keplero* and the *mouth curve*, see examples in Figure 2. Anyway, our library also includes basic families of curves like straight lines, circles and ellipses. Finally, we highlight that additional families and combinations of curves are possible [Shi95], following a curve analysis method similar to that described in the reminder of this Section.

The curves shown in Figures 1 and 2 have been identified in order to recognise feature curves corresponding to petal, or flowers in floral band decorations (*citrus, geometric petal of type A and B, m-convexities curves*), to anatomical characteristics like eyes and mouths (*citrus, geometric petal of type A and B, mouth curves*), symbols, logos or geometric decorations present in human arte-

facts like plasters or friezes (*egg of Keplero, Archimedean spiral, m-convexities, Lamet, lemniscate of Bernoulli*).

In each family of curves $\mathcal{F}$, we look for the curve $\mathcal{C}_\lambda$ that better fits a set of points $\mathcal{P}$. The set of points $\mathcal{P}$ is supposed to represent a curve or some of its parts. For each curve in our library, we highlight the parameters that drive the HT and we show how the knowledge of the main geometric characteristics of each family allows us to a-priori estimate these parameters in which λ varies for the set $\mathcal{P}$. Indeed, we use the bounding box of the set $\mathcal{P}$ to evaluate which curve parameters would generate a curve tangent to the bounding box. Then, we consider these parameters as a kind of centre of the parameter space: we automatically select a window of the parameter space and opportunely grid it to detect the value of λ corresponding to the best fitting curve. Note that, according to the notation in Section 3, λ is a vector of parameters: in the examples listed in this Section it assumes dimension one or two (i.e., one or two scalar parameters). When convenient, we use a polar representation instead of a traditional Cartesian one.

In the following, we list the curves included in our system. For each curve we show its equation, its parameters and a representation with specific parameters.

- The *citrus curve* has the Cartesian equation

$$a^4b^2y^2 + (x - \frac{a}{2})^3(x + \frac{a}{2})^3 = 0,$$

with a and b real parameters (with respect to the previous notation $\lambda = (a,b)$, we will omit the double notation in the following). It is a symmetric and limited curve, contained in the rectangle $[-\frac{a}{2}, \frac{a}{2}] \times [-\frac{a}{8b}, \frac{a}{8b}]$. An example is provided in Figure 1(a)-(b). The parameters a and b can be estimated through the size of bounding box of the set of points $\mathcal{P}$. Denoting, respectively, $D1$ and $D2$ the length of the horizontal and vertical edges of the bounding box of $\mathcal{P}$, the parameters a and b are limited by the relation $a = D1$ and $b = \frac{a}{4D2}$. The parameter b determines how the curve *squeezes* along the y axis.
- The *Archimedean spiral* is expressed in polar coordinates in the form

$$\rho = a + b\theta,$$

with a and b real parameters (see Figure 1(c)-(d)). This curve is connected and not limited, with a singularity at the point $(a,0)$. Two consecutive turnings of the spiral have a constant separation distance which is equal to $2\pi b$ and, therefore, the k-th turning of the spiral is contained in a region bounded by two concentric circles of radius $a + 2(k-1)\pi b$ and $a + 2k\pi b$. Therefore the parameters a and b depend on the length of the edges of the bounding box of the set $\mathcal{P}$ and the number of turnings of the spiral k.
- The *Lamet curve* has the Cartesian equation

$$bx^m + a^m y^m = a^m b,$$

with $a, b \in \mathbb{R}_{>0}$ and $m \in \mathbb{N}$. It is a curve of degree m and represents a rectangle with rounded corners, which increases when m grows. This curve is connected, closed and equipped with two symmetry axes (the x axis and the y axis). Furthermore, it is contained in a rectangle of edges $[-a,a] \times [-b^{\frac{1}{m}}, b^{\frac{1}{m}}]$. An example is provided in Figure 1(e)-(f). In the applications, the length of the edges of the bounding box of the set $\mathcal{P}$ allows the computation of the parameters a and b as $a = \frac{D1}{2}$ and $b = (\frac{D2}{2})^m$.
- The *m-convexities curve* is defined in polar coordinates by the equation

$$\rho = \frac{a}{1 + b\cos(m\theta)}$$

with $a, b \in \mathbb{R}_{>0}$, $b < 1$ and $m \in \mathbb{N}_+$, $m \geq 2$ (see Figure 1(g)-(h)). The parameter a plays the role of the scale factor, while b regulates the fineness of the convexities. This curve is connected and bounded. It has m symmetry axes and it is contained in the region between two concentric circumferences of radius $\frac{a}{1+b}$ and $\frac{a}{1-b}$. To estimate the parameters, we use the maximum and the minimum radius (respectively, $R2$ and $R1$) of the set $\mathcal{P}$ written in polar coordinates.
- The *geometric petal curve* of type *(A)* has a polar equation in the form

$$\rho = a + b\cos^{2n}\theta$$

with $n \in \mathbb{N}_+$ e $a, b \in \mathbb{R}$, which for our purposes is reduced to the case $b = -a$. It is a bounded and symmetrical curve, with a singularity in the origin, and it is completely contained in a circle of radius $\sqrt{2a}$. An example is provided in Figure 1(i)-(l)). Replacing a with c^2, we get the curve in Cartesian coordinates with the usual substitution $\rho = \sqrt{x^2+y^2}$ and $\cos(\theta) = \frac{x}{\sqrt{x^2+y^2}}$ of equation

$$(x^2+y^2)^{2n+1} - c^4[(x^2+y^2)^n - x^{2n}]^2 = 0.$$

This curve is bounded by the rectangle $[-\frac{2n}{2n+1}c^2\sqrt[2n]{\frac{1}{2n+1}}, \frac{2n}{2n+1}c^2\sqrt[2n]{\frac{1}{2n+1}}] \times [0, c^2]$. In this case, it is convenient to use the equation in polar form. Remembering the substitution of a with c^2, a limit for the parameter a is given by the length of the vertical edge of the bounding box of the set $\mathcal{P}$, therefore $a = D2$.
- The *geometric petal* of type *(B)* has polar equation

$$\rho = a + b\cos 2n\varphi$$

with $a > 0$, $b > 0$ and $n \in \mathbb{N}$. This curve is contained in a circle with radius $a + b$ and the origin is the center of symmetry, that becomes a singular point if $a \leq |b|$ (see Figure 2(a)), while if $a > |b|$ there are no singularities (see Figure 2(b)). As for the *m-convexities curve*, we have used the maximum and the minimum radius (respectively, $R2$ and $R1$) of the points written in polar coordinates to estimate a and b. Note that this family of curves is able to deal with an even number of petals, differently from the *m-convexities* one. Anyway, we include this function in our library because we experimentally verified that the computational time for this curve is the half of the *m-convexities curve*.
- The *lemniscate of Bernoulli* is an algebraic curve in the form of a lying eight (see Figure 2(c)-(d)). Its Cartesian equation is

$$(x^2+y^2)^2 = 2a^2(x^2-y^2)$$

with a positive real parameter. It is a symmetric and bounded curve, contained in the rectangular region $[-\sqrt{2}a, \sqrt{2}a] \times [\frac{-a}{2}, \frac{a}{2}]$. So the parameter a can be estimated as $a = \frac{\sqrt{2}}{2}D1$. For convenience, in this case we adopt the equation of the curve in

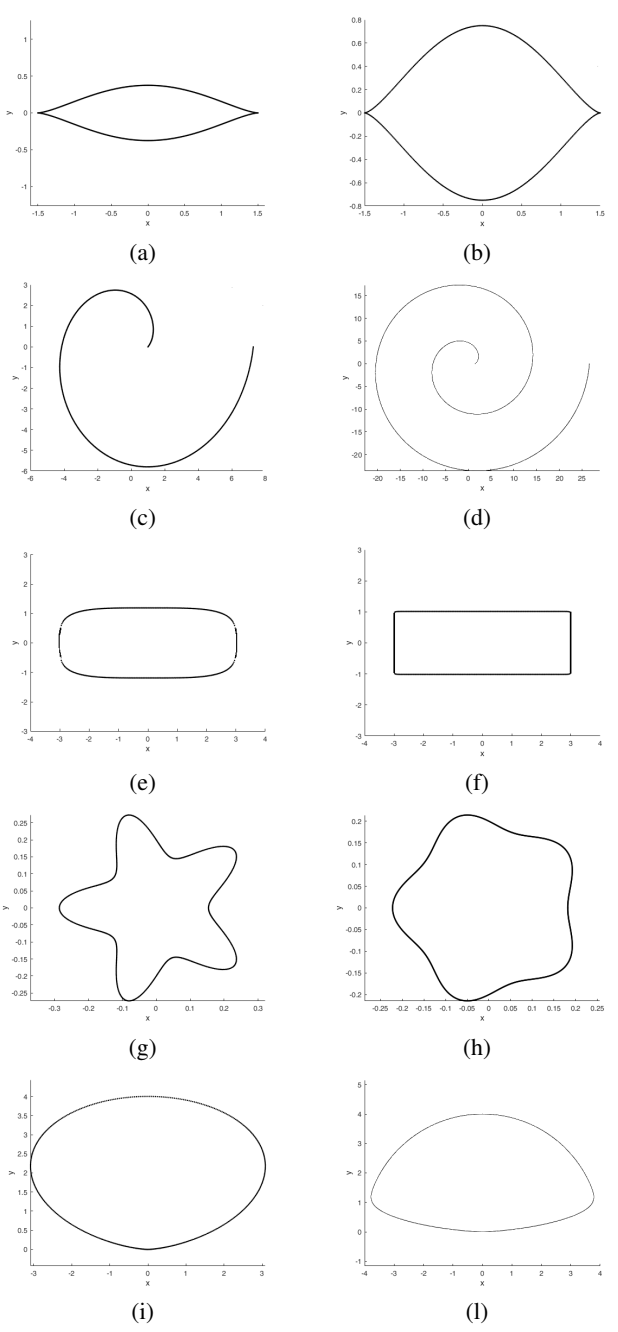

Figure 1: *Examples of mathematical curves represented with different parameters values: a citrus curve with $a = 3$ and (a) $b = 1$, (b) $b = 1/2$; an Archimedean spiral with $a = 1$ and (c) $b = 1$, $k = 1$ (d) $b = 2$, $k = 2$; a Lamet curve with $a = 3$, $b = 2$ and (e) $m = 4$, (f) $m = 50$; a m-convexities curve with $a = 0.2$, $m = 5$ and (g) $b = 0.3$, (h) $b = 0.1$; a geometric petal of type (A) with $a = 4$, $b = -4$ and (i) $n = 1$, (l) $n = 50$.*

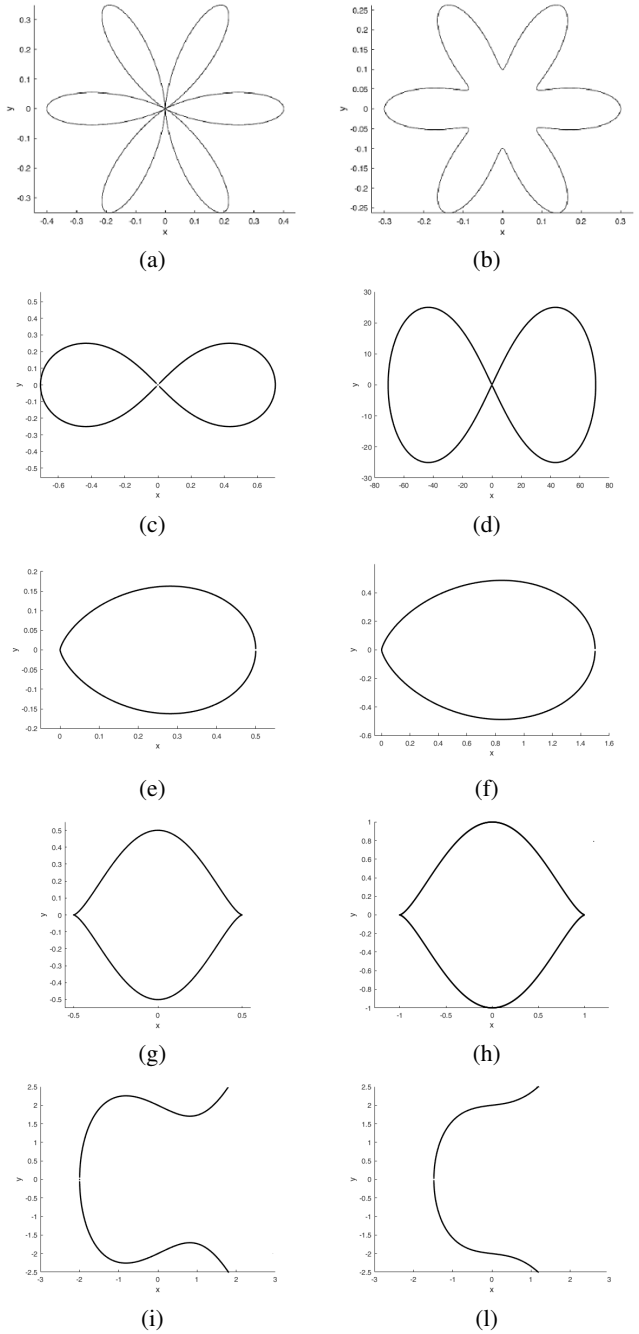

Figure 2: *Examples of mathematical curves represented with different parameters values: a geometric petal of type (B) with $a = 0.2$, $n = 3$ and (a) $b = 0.2$, (b) $b = 0.1$; a lemniscate of Bernoulli with (c) $a = 2$ and (d) $a = 50$; an egg of Keplero with (e) $a = 0.5$ and (f) $a = 1.5$; a mouth curve with (g) $a = 0.5$ and (h) $a = 1$; an elliptic curve with $b = 4$ and (i) $a = -2$, (l) $a = 1/2$.*

polar form

$$\rho^2 = 2a^2 \cos 2\theta.$$

- The *egg of Keplero* has the Cartesian equation

$$(x^2+y^2)^2 = ax^3$$

with $a \in \mathbb{R}$, $a > 0$. It is symmetric respect to the $x-$axis and bounded, in particular it is contained in the rectangle $[0,a] \times [-\frac{3\sqrt{3}}{16}a, \frac{3\sqrt{3}}{16}a]$. Therefore, the value of the parameter a coincides with the length of the horizontal edge of the bounding box of the set $\mathcal{P}$. An example is provided in Figure 2(e)-(f).

- The *mouth curve* has the Cartesian equation

$$a^4y^2 = (a^2-x^2)^3,$$

whit a real positive parameter which is half the length of the mouth. It is a symmetric and bounded curve, contained in the square $[-a,a] \times [-a,a]$. Even in this case, the parameter a is estimated through the length of the edges of the bounding box of the set $\mathcal{P}$. This curve has a shape similar to that of the citrus curve, but it extends more along the $y-$axis (see Figure 2(g)-(h)).

- The *elliptic curve* has the Cartesian equation

$$y^2 = x^3 + ax + b$$

with a and b real parameters. It is symmetric with respect to the x-axis and unbounded. We consider only the case the curve is not singular, i.e., when the determinant $4a^3 + 27b^2$ is positive. An example is provided in Figure 2(i)-(l). The parameters a and b have well specific geometric interpretations: the parameter b is the square of the ordinate of the intersection points of the curve with the y-axis (points $(0, \pm D)$), while the parameter a appears in the abscissas of the points of maximum of the semi-curve $y = \sqrt{x^3+ax+b}$ (point (x_{max}, y_{max})).

Table 1 lists the curves so far described, showing their equation, its parameters, together with their estimation and space.

Combining the simple curves listed in this Section, it is possible to recognise complex patterns made of several elements. In these patterns more families of curves or more occurrences of the same family might be present, possibly with different parameters. We deal with multiple occurrences of curves in two different ways. The first approach builds a new family of curves in terms of the product of the equations of the corresponding curve families. Some examples are provided in Section 5. The use of the curve product is very simple and yields a strategy to enrich the dictionary of curves. The idea is that the simultaneous detection of two or more curves limits the ambiguities; for instance, combining the circle at the centre of a rosette with the petals in a pattern with flowers and circles. The second method looks for the rules and parameters that characterise a pattern and replicates them, for instance repeating a pattern by translation, reflection or rotation rules. In this case, we learn the parameters that characterise a single pattern and the rules for their aggregation and repetition. In Section 5 an example of recognition of a pattern by translation is shown.

5. HT recognition on surface models

The curve recognition method is independent of the technique adopted to select the characteristic points and the data representation (we can accept both meshes or point clouds). As a working assumption, we assume that the model can be locally represented by an explicit function and therefore locally flattened. We refer to [CGL*08, LZH*07] for an overview on methods for extracting characteristic points and to the SHREC19 benchmark for an evaluation of methods for feature curves estimation [MGM*19]. In this Section, we briefly summarise the main steps of the algorithm for the HT-based curve recognition and show some examples on surface models.

5.1. Curve recognition algorithm

The method works in four main steps, namely the extraction of the characteristic points; the projection of the potential characteristic points on a plane; the HT curve recognition; and a recognition criterion based on the quality of the approximation.

The method requires as input a family of curves and some thresholds to assess which points are significant or not, to establish if a set of characteristic points can be grouped or not and to assess the quality of the recognition result.

1. *Extraction of the characteristic points*. The first step of the algorithm consists in the extraction of characteristic points through the study of geometric properties of the model. To identify these points we adopt the mean curvature. The mean curvature is evaluated with the curvature estimation based on normal cycles [CSM03] in case of meshes; it is approximated with the polynomial fitting of osculating jets [CP03] in case of point clouds. Depending on the type of feature (if a ridge or a groove) we select the high or low curvature values through the analysis of a histogram of the curvature distribution and keeping the points corresponding to the histogram queues (the size of the queue is an input parameter). The points are then grouped into connected components through a clustering operation. Many clustering algorithms are available in the literature [ELL09], in our implementation we use the method *Density-Based Spatial Clustering of Application with Noise* DBSCAN, [EKSX96], which aggregates nearby points with a certain density and eliminates the isolated ones considering them noise. This clustering choice permits the aggregation into group of points of even irregular shape but also aggregate curves that intersect; in the latter case, another clustering strategy could be adopted or a composite curve should be considered.
2. *Projection on the regression plane*. This step consists in reducing the three-dimensional problem to a two-dimensional one, projecting each single cluster on a plane. Even if every point on a surface admits a neighbourhood that is homeomorphic to a disk, this method holds only for curves that can be flattened. In our implementation, we automatically project each cluster in its regression plane using the Matlab function *regress*; in case the estimation of the normal of a cluster is numerically unstable, an alternative method for the plane approximation is the RANSAC algorithm [FB81]. Then, the minimal bounding box is used to determine its main axes and estimate its size. Subsequently, the points of each cluster are translated and/or rotated in order to place them in the default position of the curve family selected for the recognition.
3. *Recognition of the characteristic curve*. The generalised HT

Table 1:

curve	equation	parameters	parameters estimation	parameters space
citrus curve	$a^4b^2y^2+(x-\frac{a}{2})^3(x+\frac{a}{2})^3=0$	$a,b\in\mathbb{R}$	$A=D1$ and $B=\frac{D1}{2D2}$	$[A-\delta,A+\delta]\times[B-\delta,B+\delta]$
Archimedean spiral	$\rho=a+b\theta$	$a,b\in\mathbb{R}$	$A=\frac{\lvert(2-4k)D1+(4k-1)D2\rvert}{2}$ and $B=\frac{D1-2A}{(4k-1)\pi}$	$[A-\delta,A+\delta]\times[B-\delta,B+\delta]$
Lamet curve	$bx^m+a^my^m=a^mb$	$a,b\in\mathbb{R}_{>0}$	$A=\frac{D1}{2}$ and $B\frac{D2}{2}^m$	$[A-\delta,A+\delta]\times[B-\delta,B+\delta]$
m-convexities curve	$\rho=\frac{a}{1+b\cos(m\theta)}$	$a,b\in\mathbb{R}_{>0}$ with $b<1$	$A=\frac{2R1R2}{R1+R2}$ and $B=\frac{R2-R1}{R2+R1}$	$[A-\delta,A+\delta]\times[B-\delta,B+\delta]$
geometric petal (type (A))	$\rho=a-a\cos^{2n}\theta$	$a\in\mathbb{R}$	$A=D2$	$[A-\delta,A+\delta]$
geometric petal (type (B))	$\rho=a+b\cos 2n\varphi$	$a,b\in\mathbb{R}_{>0}$	$A=\frac{R2+R1}{2}$ and $B=\frac{R2-R1}{2}$	$[A-\delta,A+\delta]\times[B-\delta,B+\delta]$
lemniscate of Bernoulli	$(x^2+y^2)^2=2a^2(x^2-y^2)$	$a\in\mathbb{R}_{>0}$	$A=\frac{\sqrt{2}}{2}D1$	$[A-\delta,A+\delta]$
egg of Keplero	$(x^2+y^2)^2=ax^3$	$a\in\mathbb{R}_{>0}$	$A=D1$	$[A-\delta,A+\delta]$
mouth curve	$a^4y^2=(a^2-x^2)^3$	$a\in\mathbb{R}_{>0}$	$A=\frac{D1}{2}$	$[A-\delta,A+\delta]$
elliptic curve	$y^2=x^3+ax+b$	$a,b\in\mathbb{R}$	$A=3x_{max}^2$ and $B=D^2$	$[A-\delta,A+\delta]\times[B-\delta,B+\delta]$

technique ([TB14], [BMP13]) is applied to each cluster. The parameters of the curve that best approximates the given profile is found through the following voting procedure:

- Fix a region T of the parameter space by studying particular characteristics of the family $\mathcal{F}$ (see Section 4). The region T is discretized into cells, which are uniquely identified by the coordinates of their centre.
- Define a multi-matrix A, in which at each entry corresponds a cell of the discretization. The value of each entry increases by 1 each time the HT of a cluster point intersects the corresponding cell.
- Search for the cell corresponding to the maximum value of the matrix A and the coordinates of the cell centre correspond to the parameters of the recognised curve.

Finally, the points in the cluster that are close to the curve recognised for less than a user defined threshold (ε) are selected. In this step, to evaluate the intersection of Hough transforms with cells, we adopt the method described in [TB14] and we use the library *CoCoA* [ABL]. The strategy of the voting procedure allows the method to find the curve even if the data are partial or incomplete.

4. *Termination criterion.* To determine if the curve identified at the previous step satisfactorily approximates a set of characteristic points $\mathcal{P}$, we use the notion of *Goodness of Fit* (GoF) introduced in [MPCB15b]. Formally, for each point $P\in\mathcal{P}$ let us consider the Euclidean distance d from the curve defined as follows: $\mathcal{C}_\lambda$ recognised in the step 3:

$$d(P,\mathcal{C}_\lambda)=inf_{P_c\in\mathcal{C}_\lambda}\parallel P_c-P\parallel_2.$$

Then the GoF is defined as:

$$GoF:=\frac{d_1+d_2+...+d_{|\mathcal{P}|}}{|\mathcal{P}|},$$

where $|\mathcal{P}|$ is the cardinality of the set $\mathcal{P}$. The curve recognised is a good approximation of the profile outline if the value of GoF is smaller than a given threshold (for example, an automatic way to fix such a threshold is to select the 10% textcolorredof the lower curve parameter).

Finally, the translation and rotation operations are done backwards to identify back on the original model the initial coordinates of the feature points recognised.

The algorithm returns the parameters of the recognized curve and the vertices of the model that are closest to the curve identified by our method.

Similarly to the classical HT, the cost of the HT recognition algorithm is dominated by the size of the discretization of the region of the parameters. Such a discretization consists of $M=\prod_{k=1}^{t}J_k$ elements, where t is the number of parameters (in the curves proposed in this paper, $t=1,2,3$) and J_k is the number of subdivisions for the kth parameter. Moreover, as described in [TB14], we need to evaluate, once for each curve, the symbolic expression of the Jacobian, the Moore-Penrose pseudo-inverse and Hessian matrices which have the same order of complexity of M and, therefore, the cost of the HT-based recognition is $O(M)$ for each curve.

5.2. Experimental results

Our method has been tested on a set of models collected from the web and various repositories, in particular the benchmark proposed in the *SHape REtrieval Contest SHREC'19* (SHREC'19) track on feature curve extraction [MGM*19], the VISIONAIR shape repository [VIS15], the STARC repository†, and the ornaments in the Regency collection. The original models of the ornaments from the Regency collection are courtesy of the prof. K. Rodriguez Echavarria. All models are represented as triangle meshes or point clouds.

Figure 3 shows some examples of recognition of different patterns using the single mathematical curves listed in Section 4. In these examples, the characteristic points are extracted using different thresholds of the mean curvature. In the right column we show

† http://public.cyi.ac.cy/starcRepo/

the projected points and the curve that best approximates the profile $\mathcal{P}$ they outline. The red points are the closest to the curve with respect to the threshold ε, as detailed in Section 5.1. Backward operations are performed on these points to obtain the initial coordinates of the corresponding vertices. The outcome of the recognition algorithm is shown in the next column: the recognised vertices are highlighted on the model. The models in Fig. 3(a-b,f-i) have a known unit of measure (millimetres): depending on the curve, the parameters a and b give a precise estimation of the size of the curve; details on the curves recognised and their parameters are provided in the caption of Figure 3.

Figure 4 shows some examples of recognition of multiple elements with a product curve. In these examples, we use families of curves whose equation is defined by the product of other curve families. For instance, to recognise the fret in Figure 4(a), we used a family defined as the product of eight straight lines. Considering the minimal bounding box of the characteristic points and fixing M as the length of the horizontal edge of the bounding box, the product curve becomes:

$$xy(x-M)(y-a)(x-\frac{M}{2})(y-a+M-b)(x-b)(y-M+b)=0$$

and we found the parameters $a = 24.9180$ and $b = 24.8250$. Since the coordinates of these models are in millimetres, we can also deduce that the width of the square-like spiral is approximately 2.5 centimetres. Figure 4(b) shows the outcome of the recognition of the flower of a rosette in all its parts in terms of the product of a circle and a geometric petal of type (B) with $a > |b|$. In this case, the family we used is defined by the equation

$$(\rho - a - b\cos 2n\varphi)(\rho - r) = 0$$

and we found the parameter $a = 21.7050$, $b = 4.4950$ and $r = 4.6560$. Note that, differently from the other examples shown in this work, in this case the curve product has three parameters (namely a, b and r). Similarly to the square-like pattern, from the parameters a and b we deduce that the radius of the entire floral decoration is approximately 26 centimetres and the radius of the central circle is approximately 4.6 centimetres. Finally, Figure 4(c) represents a reproduction of a symbol known as the *'Third Paradise'* by Michelangelo Pistoletto [‡] artificially inserted on the 3D model of a vase. We recognised this symbol with a product of two eggs of Keplero and one mouth curve, so with the family of equation:

$$\begin{aligned}(((x+a)^2+y^2)+b(x+a)^3)\ (a^4y^2-(a^2-x^2)^3)\\ (((x-a)^2+y^2)^2-b(x-a)^3)=0.\end{aligned}$$

The curve parameters that best approximate this set of points are $a = 0.4240$ and $b = 0.4690$.

We have tested our method on an Intel Core i7 processor (at 4.2 GHz) with 32Mb of RAM. The timing of the recognition of the simple curves is in the order of 10 seconds, except for the elliptic and the Archimedean spiral, which take about 30 seconds and 90 seconds, respectively. For the compound curves in Figure 4, the timing of the recognition is in the order of 1 minute.

The HT can recognise more occurrences of a curve on the model. Two examples are provided in Figure 5: the two eyes of the statue in Figure 5(a) are recognised with two citrus curves having parameter $a = 28.2360$ and $a = 28.151$, respectively. Similarly the keyboard buttons in Figure 5(b) are recognised with circles of radius $r = 1.0750$ or Lamet curves whose parameters are $a = 1.303$ and $b = 0.003$ on average. For each button, two curves are recognised with identical parameters; in this case, the characteristic points are selected using both high and low values of the mean curvature.

Figure 5(c) illustrates the recognition of a repeated pattern. The pattern is made of two elements: the oval-like curve shown in Figure 3(f) and the half of a mouth-like curve in Figure 3(g). The pattern is then horizontally repeated to form a frieze of four elements. The parameter a of the mouth-like curve corresponds to half of the curve width, therefore the amplitude of the curve is $2a$, that physically corresponds to approximately 11 centimetres (in this example $a = 55.67$ and the coordinates are millimetres). Then, we use the parameter $2a$ obtained for the recognition of a single element to infer the entity of the horizontal translation and to recognise the entire moulding. In practice, we recognise the entire frieze by translating the pattern curves of $2a$. In Figure 5(c), we represent in yellow the points corresponding to the mouth-like curve and in blue the points coming from the oval ones.

In Figure 5(d), we show the outcome of both curve recognition and comparison. The model comes from the SHREC'19 contest [MGM*19]. In that benchmark, four models with spiral-like patterns were proposed and, besides curve recognition, the challenge was to propose a similarity distance between the curves recognised. A possible way to address this challenge with a HT-based curve recognition method is to compare the curve parameters of the curves that belong to the same family but are (possibly) recognised on different models or parts. As a very simple measure for parameter comparison we consider the ℓ_1 norm as the curve distance. Even adopting such a simple measure, it is possible to obtain a number of results. For instance, the model in Figure 5(d) is the union of two SHREC'19 models; such a union is decided on the basis of all the distances among the spiral-like curves and keeping the smallest one. Numerically, in this specific case, the parameters of the two curves are respectively $a = 2.7300$, $b = 1.9660$ and $a = 3.5550$, $b = 2.6970$ and the distance is 0.0638. Looking at the union of the two models, it is possible to argue that they belong to the same moulding.

6. Conclusions

This paper describes a method for the recognition of characteristic curves on 3D shapes. It makes use of a dictionary of template curves, which can be composed or aggregated. We approximate these characteristic curves with mathematical functions using a generalisation of the HT.

Our approach is quite innovative and different from the numerous existing techniques that fit feature curves with splines and cannot give their global equation. The only methods which exhibit a certain similarity with ours are the *natural 3D spiral* [HT11] and the *3D Euler spiral* [HT12]. However, these methods can be used only to identify these specific curves and do not have the completeness of our system.

‡ http://www.pistoletto.it/eng/crono26.htm

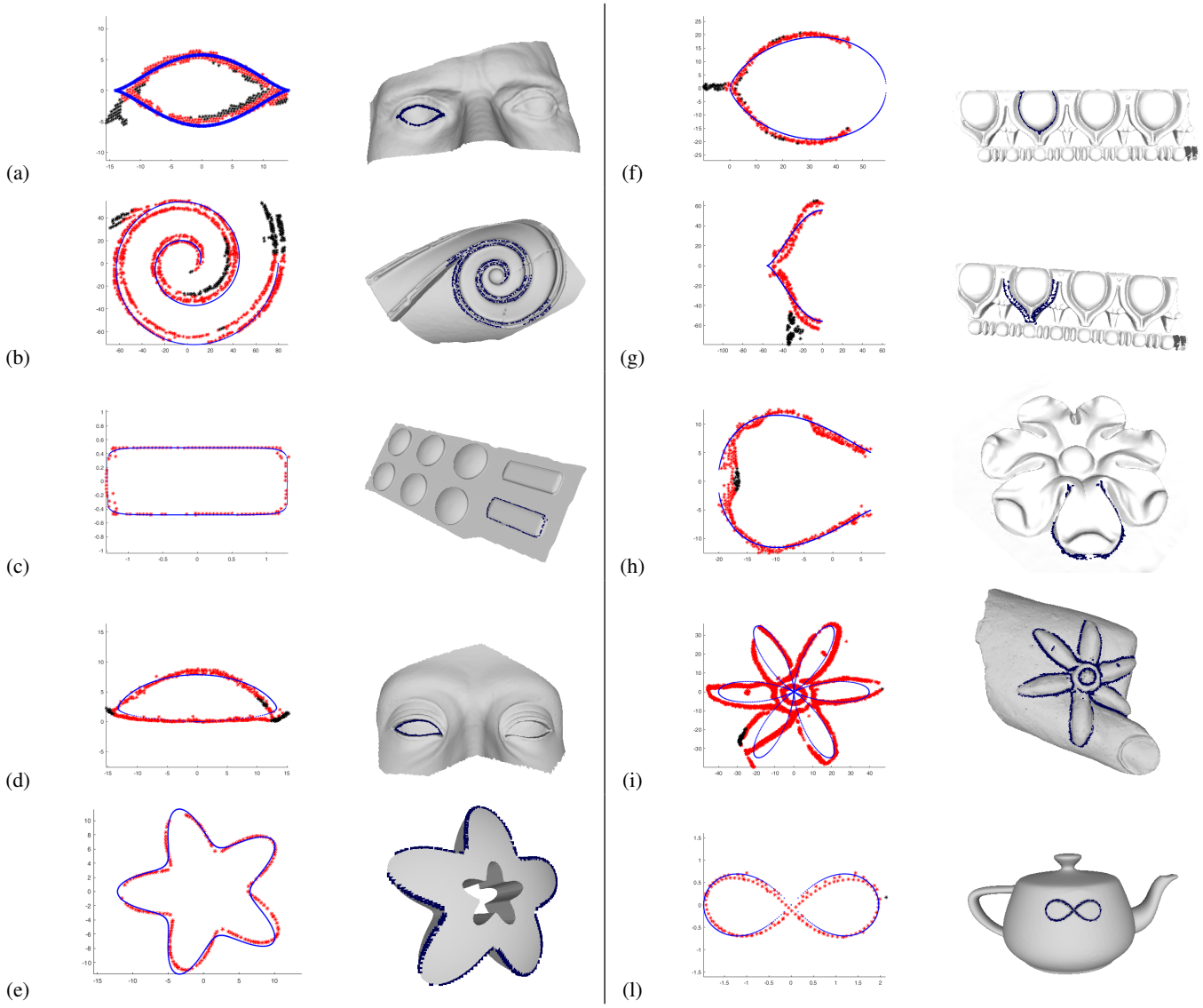

Figure 3: *Recognition of various characteristic curves. The recognised curves and the characteristic points are shown, highlighting in red the points that are the closest to the curve. The vertices corresponding to these points are then represented back on the model. The curves and their parameters are: (a) a citrus curve with $a = 28.2360$ and $b = 0.6147$, (b) an Archimedean spiral with $a = 10.6670$ and $b = 5.5060$, (c) a Lamet curve with $a = 1.3030$, $b = 0.0030$ and $m = 8$, (d) a geometric petal curve of type (A) with $a = 7.8490$, $c = 0.5593$ and $n = 50$, (e) a m-convexities curve with $a = 8.5090$, $b = 0.3000$ and $m = 5$, (f) an egg of Keplero with $a = 8.8660$, (g) a mouth curve with $a = 55.6700$, (h) an elliptic curve with $a = 28.3790$ and $b = 76.0490$, (i) a geometric petal curve of type (B) with $a = 19.9590$ and $n = 3$, (l) a lemniscate of Bernoulli with $a = 1.3890$. The models in the first column ((a), (b), (c), (d), (e)) have been proposed in the SHREC'19 contest, in particular the original models in (a) and (e) come from the VISIONAIR shape repository [VIS15]. The models (f), (g) and (h) come from the Regency collection of 3D ornaments, the model in (i) comes from the STARC repository and the model in (l) has been artificially created with Blender [Ble18].*

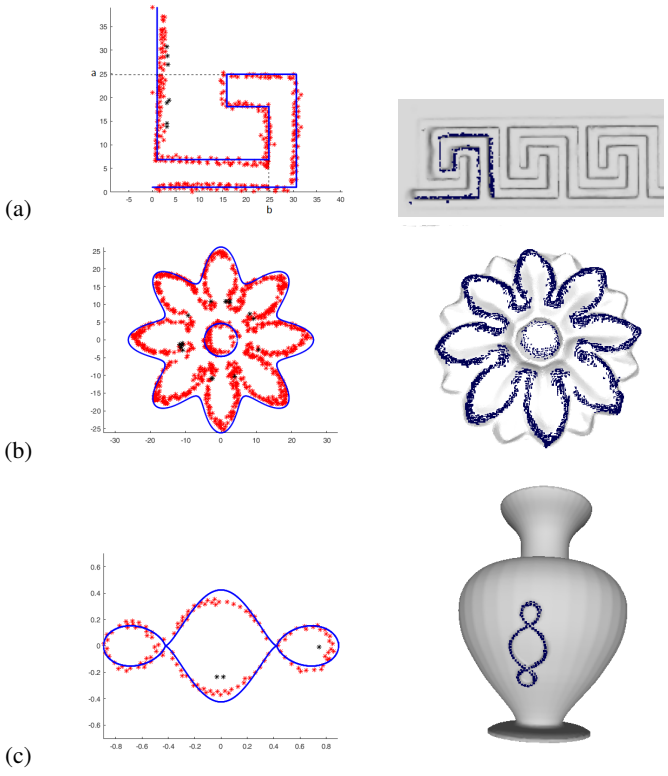

Figure 4: *Examples of recognition of complex patterns through the product of curves composing it.*

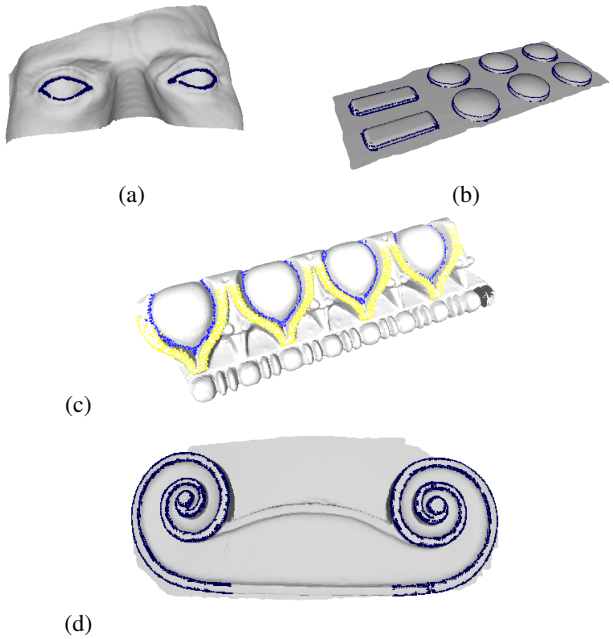

Figure 5: *Examples of recognition of complex patterns through the use of repetition rules.*

Indeed the system we have implemented is open and allows new additions of curves in the dictionary of functions already available. For planar curves large dictionaries of curves exist [Shi95], while such a richness of curve templates does not exist for spatial curves. Moreover, our method supports the definition of product curves and new rules of composition or aggregation of characteristic curves to recognise compound patterns.

Note that our method targets the curve recognition problem rather than an exact curve approximation, where curve approximation methods rooted on curve splines [CRS18] are expected to achieve a very precise fitting but lose the global view of the curve element.

Indeed, the most appealing property of the HT is its ability of recognising a pattern of curves in its entirety, even in presence of noise and partial data: this implies that the HT is naturally suitable for shape completion, annotation and multiple curve and pattern comparison.

Through the analysis of the curve parameters it is possible to find similarities among the curve elements even in objects that are different in terms of their overall shape, structure and function as preliminarily shown for some of the models proposed in [MGM*19].

Moreover, the method provides an efficient localisation of a curve on the surface and an explicit estimation of the size of a shape feature because the parameters are strictly related to the physical measure of the curve. This information is therefore suitable to support the automatic annotation of the elements and the patterns on 3D digital models. In the case of complex patterns like the model shown in Figure 5(c) it is possible to localise and label with their dimension the points corresponding to a single curve, the points corresponding to the combination of the two pattern curves and, also, the complete frieze, distinguishing it from the rest of the 3D model.

Acknowledgements

This work has been developed in the CNR IMATI research activities DIT.AD004.028.001 and DIT.AD021.080.001. The authors thank Elia Moscoso Thompson for his help in creating the examples of Figures 3l and 4c, and Michela Spagnuolo, for her support and the helpful discussions on this topic.

References

[ABL] ABBOTT J., BIGATTI A. M., LAGORIO G.: CoCoA-5: a system for doing Computations in Commutative Algebra. Available at http://cocoa.dima.unige.it. 6

[APM15] ANDREADIS A., PAPAIOANNOU G., MAVRIDIS P.: Generalized digital reassembly using geometric registration. In *Digital Heritage* (2015), vol. 2, pp. 549–556. 1

[Bal81] BALLARD D. H.: Generalizing the Hough transform to detect arbitrary shapes. *Pattern recogn. 13*, 2 (1981), 111–122. 2

[Ble18] BLENDER ONLINE COMMUNITY: *Blender - a 3D modelling and rendering package*. Blender Foundation, Blender Institute, Amsterdam, 2018. URL: http://www.blender.org. 8

[BMP13] BELTRAMETTI M., MASSONE A., PIANA M.: Hough transform of special classes of curves. *SIAM J. Imaging Sci. 6*, 1 (2013), 391–412. 1, 2, 6

[BR12] BELTRAMETTI M. C., ROBBIANO L.: An algebraic approach to Hough transforms. *J. of Algebra 37* (2012), 669–681. 1, 2

[CGL*08] COLE F., GOLOVINSKIY A., LIMPAECHER A., BARROS H. S., FINKELSTEIN A., FUNKHOUSER T., RUSINKIEWICZ S.: Where do people draw lines? *ACM Trans. Graph. 27*, 3 (Aug. 2008), 1–11. 1, 5

[CP03] CAZALS F., POUGET M.: Estimating differential quantities using polynomial fitting of osculating jets. In *EG/ACM SIGGRAPH Symp. on Geometry Processing* (2003), pp. 177–187. 5

[CRS18] CONTI C., ROMANI L., SCHENONE D.: Semi-automatic spline fitting of planar curvilinear profiles in digital images using the Hough transform. *Pattern Recogn. 74*, C (Feb. 2018), 64–76. 2, 9

[CSM03] COHEN-STEINER D., MORVAN J.-M.: Restricted Delaunay triangulations and normal cycle. In *ACM Symp. on Computational Geometry* (New York, NY, USA, 2003), ACM, pp. 312–321. 5

[DH72] DUDA R. O., HART P. E.: Use of the Hough transformation to detect lines and curves in pictures. *Commun. ACM 15*, 1 (1972), 11–15. 2

[DIOHS08] DANIELS II J., OCHOTTA T., HA K. L., SILVA T. C.: Spline-based feature curves from point-sampled geometry. *The Visual Computer 24*, 6 (2008), 449–462. 1

[EKSX96] ESTER M., KRIEGEL H. P., SANDER J., XU X.: A density-based algorithm for discovering clusters in large spatial databases with noise. In 2nd *Int. Conf. Knowledge Discovery and Data Mining* (1996), AAAI Press, pp. 226–231. 5

[ELL09] EVERITT B. S., LANDAU S., LEESE M.: *Cluster Analysis*, 4th ed. Wiley Publishing, 2009. 5

[Far93] FARIN G.: *Curves and Surfaces for Computer Aided Geometric Design (3rd Ed.): A Practical Guide*. Academic Press Professional, Inc., San Diego, CA, USA, 1993. 1

[FB81] FISCHLER M. A., BOLLES R. C.: Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM 24*, 6 (June 1981), 381–395. 5

[Hou62] HOUGH P. V. C.: Method and means for recognizing complex patterns, 1962. US Patent 3,069,654. 2

[HPW05] HILDEBRANDT K., POLTHIER K., WARDETZKY M.: Smooth feature lines on surface meshes. In *EG Symp. on Geometry Processing* (2005), The Eurographics Association. 1

[HT11] HARARY G., TAL A.: The Natural 3D Spiral. *Computer Graphics Forum 30*, 2 (2011), 237–246. 1, 7

[HT12] HARARY G., TAL A.: 3D Euler spirals for 3D curve completion. *Computational Geometry 45*, 3 (2012), 115 – 126. 1, 7

[KST08] KOLOMENKIN M., SHIMSHONI I., TAL A.: Demarcating curves for shape illustration. *ACM Trans. Graph. 27*, 5 (2008), 157:1–157:9. 1

[KTT99] KASSIM A., TAN T., TAN K.: A comparative study of efficient generalised Hough transform techniques. *Image and Vision Computing 17*, 10 (1999), 737 – 748. 2

[LWWS15] LI C., WAND M., WU X., SEIDEL H. P.: Approximate 3d partial symmetry detection using co-occurrence analysis. In *2015 International Conference on 3D Vision* (Oct 2015), pp. 425–433. 1

[LZH*07] LAI Y. K., ZHOU Q. Y., HU S. M., WALLNER J., POTTMANN H.: Robust feature classification and editing. *IEEE Trans Vis Comput Graph 13*, 1 (Jan 2007), 34–45. 5

[MC15] MUKHOPADHYAY P., CHAUDHURI B. B.: A survey of Hough transform. *Pattern Recogn. 48*, 3 (2015), 993 – 1010. 2

[MGM*19] MOSCOSO THOMPSON E., GERASIMOS A., MOUSTAKAS K., NGUYEN E. R., TRAN M., LEJEMBLE T., BARTHE L., MELLADO N., ROMANENGO C., BIASOTTI S., FALCIDIENO B.: SHREC'19 track: Feature Curve Extraction on Triangle Meshes. In *EG Work. 3D Object Retrieval* (2019), The Eurographics Association. 1, 5, 6, 7, 9

[MPCB15a] MASSONE A. M., PERASSO A., CAMPI C., BELTRAMETTI M. C.: Profile detection in medical and astronomical images by means of the Hough transform of special classes of curves. *Journal of Mathematical Imaging and Vision 51*, 2 (Feb 2015), 296–310. 2

[MPCB15b] MASSONE A. M., PERASSO A., CAMPI C., BELTRAMETTI M. C.: Profile detection in medical and astronomical images by means of the Hough transform of special classes of curves. *J. Math. Imaging Vis.* (2015), 296–310. 6

[OLA14] OESAU S., LAFARGE F., ALLIEZ P.: Indoor Scene Reconstruction using Feature Sensitive Primitive Extraction and Graph-cut. *ISPRS J. of Photogrammetry and Remote Sensing 90* (Mar. 2014), 68–82. 2

[PT97] PIEGL L., TILLER W.: *The NURBS Book (2Nd Ed.)*. Springer-Verlag New York, Inc., New York, NY, USA, 1997. 1

[RBM15] RICCA G., BELTRAMETTI M. C., MASSONE A. M.: Piecewise recognition of bone skeleton profiles via an iterative Hough transform approach without re-voting, 2015. 2

[Shi95] SHIKIN E. V.: *Handbook and atlas of curves*. CRC, 1995. 1, 2, 9

[SJW*11] SUNKEL M., JANSEN S., WAND M., EISEMANN E., SEIDEL H.-P.: Learning line features in 3D geometry. *Computer Graphics Forum 30*, 2 (April 2011). 1

[TB14] TORRENTE M.-L., BELTRAMETTI M. C.: Almost vanishing polynomials and an application to the Hough transform. *J. of Algebra and Its Applications 13*, 08 (2014), 1450057. 6

[TBF18] TORRENTE M.-L., BIASOTTI S., FALCIDIENO B.: Recognition of feature curves on 3D shapes using an algebraic approach to hough transforms. *Pattern Recogn. 73* (2018), 111 – 130. 1, 2

[VIS15] The Shape Repository. http://visionair.ge.imati.cnr.it/ontologies/shapes/, 2011–2015. 6, 8