SHAPE RECOGNITION USING ARTIFICIAL INTELLIGENCE

Ll. Museros[(1) (2)], M.T. Escrig[(1)], J.F. Noguera[(2)], I. Fuentes[(2)], G.Mallol[(2)]

[(1)] Cognition for Robotics Research Group (C4R2). Department of Computer Sciences and Engineering
Universitat Jaume I Castellón.
[(2)] Instituto de Tecnología Cerámica (ITC). Asociación de Investigación de las Industrias Cerámicas (AICE)
Universitat Jaume I. Castellón. Spain.

ABSTRACT

This study presents a new qualitative theory of shape recognition and its application in the ceramic industry for intelligent automation of the mosaic mural assembly process. The theory describes qualitatively the shapes of the objects, considering various characteristics: on the one hand, it considers shape boundary characteristics, such as angles, relative edge length, concavities and convexities, and types of curvature; on the other, it considers the colour of the objects. The shapes of the figures (objects) to be recognised may be regular or irregular closed polygons, which may contain curved segments, or even be completely closed curvilinear figures. In addition, all figures may contain holes. Each figure is described and represented as a character chain that contains all its distinctive qualitative characteristics (a symbolic representation is involved). This representation is used to calculate whether the shape of two figures matches.

The developed application consists of the following: given an image of various ceramic tesserae and given a vectorial design of the mosaic to be assembled, to calculate which tessera in the image matches a tessera in the design. The application not only calculates the position of the image tessera according to the design, but also calculates the rotation angle that a manipulating arm needs to travel (when picking up the tessera by its centroide) in order to place the tessera in the correct position and orientation according to the design.

1. INTRODUCTION

This paper presents the application of a new theory of shape recognition in the ceramic industry. The purpose of the application is the recognition of the tesserae that form a mosaic, thus enabling intelligent automated assembly of ceramic mosaics.

A mosaic is a decorative art form in which small ceramic tiles known as tesserae are assembled to form a predefined image.

There are two main methods of creating mosaics, commonly called the **"direct method"** and the **"indirect method"** of constructing mosaics.

The **direct method** consists of directly installing (adhering) the individual tesserae on the substrate surface. This method is the more appropriate one when the surfaces to be covered with the mosaic have a three-dimensional shape, such as a vessel, and is useful in small transportable projects. In addition, it has the further advantage of allowing one progressively to see the mosaic materialise as it is being created, so that the necessary fixing and colour adjustments in the tesserae can be made immediately. But it also has the following disadvantages:

- The artist needs to work directly on the surface to be covered, which is often impracticable during long periods of time (for example, owing to bad weather, surfaces in areas under construction, etc.).
- It is inappropriate in large-scale projects.
- It is difficult to control finished surface uniformity, a matter of particular importance when a functional surface is created, such as flooring.

If some of the foregoing features occur, the indirect method of constructing mosaics is more appropriate.

The **indirect method** of constructing mosaics consists of previously assembling and pre-arranging the tesserae on various grids or substrates that are subsequently fixed onto the surface to be clad with the mosaic. The grids are installed according to a specific plan, in order to create a particular decorative image. Each grid also follows a plan or design for arranging the individual tesserae on the grid, so that when the grids are joined they will create the image sought.

The application described in this paper is intended for application in a complete system for intelligent automation, by the **indirect method,** of constructing ceramic mosaics, i.e. the process of preassembling the tesserae in order to create different grids that then need to be fixed on the surface to be covered. To achieve this objective, an application has been designed with the architecture illustrated in Figure 1.

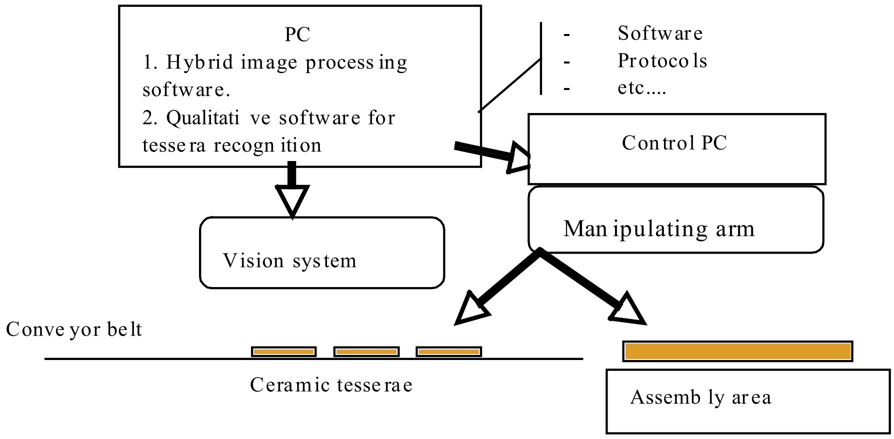


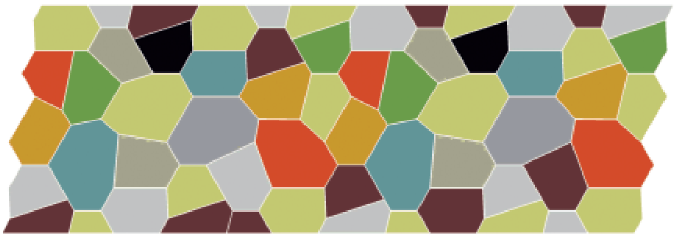
Figure 1. Architecture of the developed application.

Figure 1 shows the physical components used to achieve intelligent assembly of ceramic mosaics. The following components were involved:

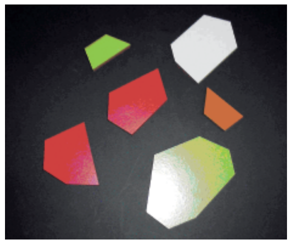
- Conveyor belt
- Vision system
- Manipulating arm with its corresponding controller
- Assembly table
- PC, containing the intelligent tools developed for this application.

Two intelligent tools were developed: a software program that allows information to be extracted from the images (captured by the vision system) of the tesserae on the conveyor belt, and a software program that recognises the tesserae and determines not just their position according to the mosaic design to be assembled, but also the rotation angle that the manipulating arm needs to travel in order to pick up the tessera with its centroide (using suction pads) and place it in the correct position and orientation (determined by the design). This last software implements a qualitative theory for object representation and recognition, a theory described in [1], which describes the shape of an object by qualitatively considering its angles, relative length of its edges, concavities and convexities, as well as the type of curvature of the boundary curves. In addition, it considers qualitatively the colour of the shape. The shapes recognised are regular or irregular closed polygons, shapes that contain curves or that are even completely curvilinear. On the other hand, the shapes to be recognised may also contain holes. For the description of the shapes with holes, qualitative topological and orientation aspects are used, with a view to relating the hole to the container. Each object is described with a character chain that comprises the distinctive qualitative characteristics of the shape (a symbolic representation is involved). This representation is used to recognise one object among others. Given the application to be developed, a qualitative approach was the most appropriate, since this type of approach allows uncertainty to be efficiently dealt with, such as for example the uncertainty associated with the fact that there are no two exactly identical tesserae, or the uncertainty associated with work under industrial conditions.

However, given the ultimate objective of the application to be developed, the foregoing software needed to interact with a vision system in order to obtain the images of the tesserae on the conveyor belt that are to be qualitatively described. In fact, the software has two inputs: on the one hand the vectorial design of the mosaic to be assembled (Figure 2a is an example) and, on the other, an image with one or more tesserae produced by the vision system (Figure 2b is an example). Therefore, it has been necessary to develop another image processing software program in order to obtain the relevant information concerning each tessera in the image, information that is then to be used to create the qualitative description of the shape of each tessera.



a)



b)

Figure 2. a) Example of the vectorial design of a mosaic to be assembled. b) Example of an image with the real tesserae in different colours and shapes to be used to assemble the mosaic design.

The rest of the paper is organised in such a way as to describe each component of the developed prototype architecture. Section 2 presents a brief overview of the state of the art in the existing qualitative theories of shape recognition. Section 3 presents the qualitative theory of shape recognition implemented for recognition of the tesserae. Section 4 describes the developed image processing software. Section 5 shows the developed complete prototype, and section 6 presents the conclusions and various ideas regarding future work.

2. STATE OF THE ART IN QUALITATIVE THEORIES OF SHAPE RECOGNITION

Given the characteristics of the objects to be determined and the uncertainty associated with the fact that the quantitative characteristics of two ceramic tiles are not exactly identical, for the development of the application it was decided to use and study in depth qualitative recognition techniques. These techniques afford several advantages in the current case because they are ideal for handling the uncertainty mentioned and, since they recognise an object by just considering its relevant characteristic and not using quantitative data, they speed up the recognition processes. Therefore, a study was undertaken of the state of the art in qualitative recognition techniques, which led to the development of an optimum theory (described in section 3) for the study of the intended application.

A brief introduction follows, first, to explain the concept of qualitative shape recognition and, secondly, to set out the state of the art.

Our environment contains objects that may be described in terms of their shape. The shape of an object consists of the description of the properties of the object's boundary. Note that an object's boundary is described as a set of points. A single point has neither dimension nor form, but once a one-dimensional curve is obtained, the curve is associated with a shape that can be described.

A purely quantitative representation of figures consists of a set of mathematical functions of the coordinate space. For example, a circumference may be described by the following mathematical function:

$$x^2+y^2=r^2$$

However, in the case of more complex figures, it is generally difficult to find a numerical function to describe the curve or surface that describes the boundary of the figure. In these cases, piecewise interpolation methods are typically used; i.e. the shape of the object is described by approximation using a set of small parts, such as straight lines or flat surfaces, for which numerical functions may be found. The set of all the functions represents the quantitative description of the object's shape. A widely used alternative to this method is to make a quantitative representation of the object's shape as a function of the pixels in the object image. In this case, however, the result is coarser or finer depending on the resolution used, since some pixels may be only partly occupied. In addition, in this case the description of the shape may vary considerably if the object appears rotated or in different positions in the pixel mesh.

In artificial vision, image processing entails high computational costs. In addition, object recognition from image processing is at present still a subject of research, because when quantitative techniques are used, for example, it is still not possible to distinguish the same chair from different viewpoints or if that chair is partly hidden.

It was decided, therefore, to use qualitative theories for describing shapes, whose use could enhance efficiency in vision recognition, since shape or environment recognition would occur solely by studying the relevant features and not by analysing each image pixel.

It is therefore useful to describe what is understood by the qualitative representation of a shape. A purely qualitative representation is provided by linguistic terms such as "round", "straight", etc. Figure 3 shows an example with the quantitative and qualitative description of a circumference.

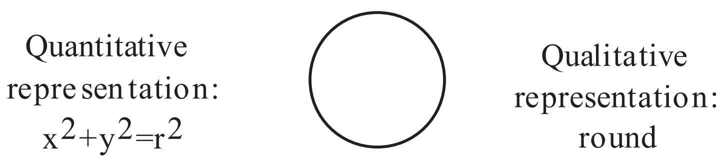


Figure 3. Examples of the quantitative and qualitative representation of a circumference.

Most of the qualitative theories developed for shape description may be classified as follows:

- **Axial representations**: these theories are based on the description of an object's axes and describe the shape qualitatively, reducing it to a "skeleton" or "axis". The axis is a flat arc reflecting some symmetry, or overall or local shape regularity. This shape may be generated from the axis if a geometric figure, termed a "generator", is moved along the axis. The generator is a constant shape that keeps a specific point (usually its centre) on the axis, but whose size and inclination may vary with respect to the axis. This group includes the studies cited in [2] and [3].

- **Primitives-based representations**: in these representations, complex objects are described as combinations of multiple simple objects or primitives. Two approaches may be distinguished in these types of representations:
    - Representations based on generalised cylinders or on "geons", which describe an object as a set of primitives plus a set of spatial connectivity relations between these. Studies in this group are cited in [4] and [5].
    - Constructive representations, which describe an object as Boolean combinations of primitive point sets [6].
- **Reference points- and projection-based representations.** These represent different aspects of the object's shape, observing it from different angles or projecting it on different axes [7],[8].
- **Topology or logic-based representations of an object's shape** [9],[10].
- **Coverings-based representations**, which describe an object's shape by covering it with simple figures, such as rectangles and spheres [11].

The theory developed for the present application may be classified as a reference points-based representation, since the theory uses information on the vertices and maximum curvature points to describe an object's shape. It may be noted that the theory developed also takes into account the relative length of the edges as well as the colour of the objects, as relevant characteristics in the recognition process. These aspects had not been considered in other studies.

3. QUALITATIVE THEORY OF SHAPE RECOGNITION

As mentioned, the theory developed in this project may be classified as a reference points-based theory, or theory based on the representative points of a shape (vertices, inflection points, and maximum curvature points). Reference points are involved because they fully specify a figure's boundary. Depending on the type of figure to be described, certain boundary points are chosen as reference points. Thus, in the case of polygons the reference points are the vertices, while in the case of curvilinear figures or curvilinear segments, the reference points are each start and end point of the curvilinear segments, and the maximum curvature point.

The qualitative description of a reference point, j, is determined using the preceding reference point, i, and the next reference point, k. Reference point order is determined by the natural cyclical order of the vertices of closed objects. Only the direction must be determined in which each reference point will be visited or described, since it shall be the same direction for all the objects to be described. In this case, the object's vertices are visited in clockwise order. The description of each reference point is given by a set of three elements, which differ when straight or curved segments are involved, as follows:

- When points (vertices) are involved that belong to straight segments, a triplet is constructed of the form <Aj, Cj, Lj>, where Aj determines the type of angle, Cj the type of convexity, and Lj the relative length of each pair of consecutive edges (the edge that reaches the vertex, *j*, and the edge that leaves this). The triplet elements may adopt the following values:

  Aj ∈ {right angle, acute, obtuse}

  Cj ∈ {convex, concave}

  Lj belongs to the Length Reference System (LRS) defined as LRS ={shorter, equal, longer}

- If points are involved that belong to curved segments, a triplet of the form <Curve, Cj, TCj> is constructed in which the *Curve* symbol determines that the node described forms part of a curved segment, Cj determines the type of convexity, and TCj determines the type of curvature.

The way in which each of these characteristics is determined is set out below, first for the straight segments and then for the curved segments.

3.1. DESCRIPTION OF REGULAR AND IRREGULAR POLYGONAL OBJECTS

To characterise the reference points, the qualitative theory of shape representation uses the Zimmermann and Freksa orientation Reference System, augmented by a circle, as a tool [12]. In the Zimmermann and Freksa orientation model, the space is divided into qualitative regions by means of a Reference System (RS). The RS is formed by an oriented line determined by two reference objects (from a point, *a*, to another point, *b*), which defines the left/right dichotomy; the perpendicular to the object *b*, which defines the first front/back dichotomy; and the line perpendicular to the object *a*, which establishes the second front/back dichotomy and defines a fine division of the space at the rear of the RS. If only a front/back dichotomy is considered (coarse reference system), the space is divided into 9 qualitative regions. If both perpendicular lines are taken into account, the space is divided into 15 qualitative regions (fine reference system). Our theory focuses on the fine RS. Figure 4a shows a representation of the fine RS, while the names of the regions are displayed in Figure 4b. The information that may be represented by this RS is the qualitative orientation of an object represented by point *c*, in regard to the RS formed by points *a* and *b*, that is, *c wrt ab*.

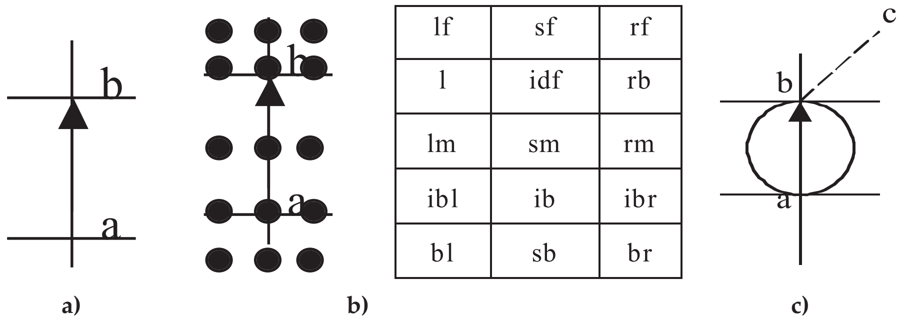


Figure 4. a) Fine RS for qualitative orientation; b) the iconic representation in which each symbol corresponds is as follows: l: left, r: right, f: front, s: straight, m: middle b: back, i: identical; c) fine RS for orientation augmented by a circle with diameter ab.

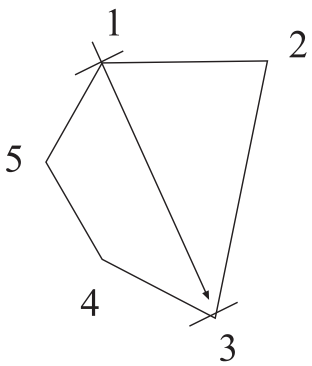


Figure 5. Example of a figure in which the qualitative description of vertex 2 is being determined, using 1 and 3 as reference vertices for the Zimmermann and Freksa orientation RS..

The central idea of the qualitative shape representation for straight line segments consists of the following: given three consecutive reference points (vertices) *i, j, k*, owing to the clockwise numbering of the object's vertices, the qualitative description of reference point *j* is determined by setting the RS between points *i* and *k*, as shown in Figure 5. In this figure, *i* is vertex 1, *j* is vertex 2, and *k* is vertex 3; the RS is set from 1 to 3.

Thus, the qualitative angle is determined using the Zimmerman and Freksa RS augmented by a circle, and certain concepts topological such as the boundary, outside and inside of the figure. In order to understand how a vertex angle is determined, the definitions of these topological concepts are therefore required.

Definition 1. The **boundary** of a figure h, termed δ**h**, is defined as follows:

- The boundary of a point is always considered empty.
- The boundary of a line is the empty set in the case of a circular line or, in other words, the 2 different end points.
- The boundary of an area is the circular line that consists of the entire area's accumulation points.

Definition 2. The **inside** of a figure h, termed $\mathbf{h^o}$, is defined as $h^o = h - \delta h$.

Definition 3. The **outside** of a figure h, termed $\mathbf{h^-}$, is defined as $h^- = \Re 2 - h$.

Taking into account the foregoing, a reference point *j* is determined as follows (Figure 6):

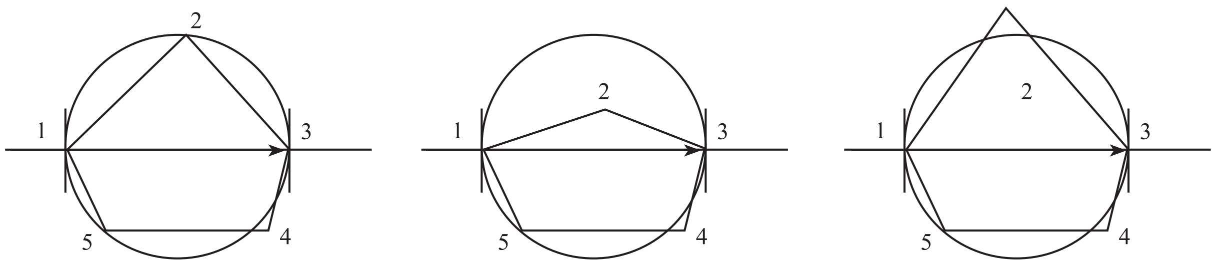
- If the reference point *j* is located exactly on the boundary of the circle that forms part of the augmented RS, then vertex *j* forms a right angle.
- If the point *j* is located on the outside of this circle, then vertex *j* forms an acute angle.
- And if point *j* is located on the inside of the circle, then vertex j forms an obtuse angle.

Formally this is denoted as follows: if the RS circle augmented by diameter ViVk is denoted as Cik, then the angle of Vertex j (Vj) is calculated using the following algorithm:

If $Vj \cap \delta Cik \neq \emptyset$ then Vj is a right angle,

But if $Vj \cap Cik^{o} \neq \emptyset$ then Vj is obtuse

Otherwise Vj is acute.



a) Right angle b) Obtuse angle c) Acute angle

Figure 6. Graphic examples that illustrate the determination of the type of angle for vertex 2: a) for a right angle; b) for an obtuse angle and c) for an acute angle.

On the other hand, to determine the convexity of point *j* pertaining to a straight segment, the RS is used again, proceeding as follows: if reference point *j* remains on the left dichotomy created by the line oriented from *i* to *k*, then point *j* is a convex vertex. This means that point *j*, in regard to the Reference System formed by *ik* (*j wrt ik*), is on one of the following orientations: left-front, left, left-middle, identical-left back, or back-left. However, if point *j* remains on the right dichotomy created by the line oriented from *i* to *k*, then point j is concave, which means that *j wrt ik* belongs to some of the following orientations: right-front, right, right-middle, identical-right back, or back-right. Since a vertex appears when the orientation of the axis changes, then the reference point *j* cannot remain exactly on the line oriented from *i* to *k*. Formally, if *Vj* represents the notation of the vertex *j*, the following may be posited:

If Vj wrt ViVk ∈ [lf, l, lm, ibl, bl] then Vj is convex.

If Vj wrt ViVk ∈ [rf, r, rm, ibr, br] then Vj is concave.

Finally, to characterise a single straight line segment it is only necessary to describe the qualitative relative length. For this a new reference system, called the relative Length Reference System (LRS), is used, which is defined by a set of labels of qualitative lengths:

LRS = {shorter, equal, longer}

The calculation of length at a reference point *j* is the length of the edge from point *i* to point *j* compared with the length of the edge that runs from *j* to *k*, using the LRS to assign the corresponding qualitative label. Only the length between consecutive straight line segments is compared. The process followed to determine this length is as follows::

- First, the length of each side is calculated using the Euclidean distance between two points, *D (Vi, Vj):*

  $D (Vi, Vj) = ((Xvj\text{-}Xvi)^2 + (Yvj\text{-}Yvi)^2)^{1/2}$

where *Vi* and *Vj* are two vertices whose coordinates *x* and *y* are denoted as Vi = (Xvi, Yvi) and Vj = (Xvj, Yvj).

- Once this Euclidean distance between vertices *i* and *j* has been calculated, termed *dij*, and between vertices *j* and *k*, termed *djk*, both lengths are compared and the corresponding LRS label is assigned as the value of the relative length to the vertex *j*. Formally:

If dij>djk → Lj = longer

But if Si dij = dik → Lj = equal

Otherwise Lj = shorter

3.2. DESCRIPTION OF OBJECTS WITH CURVED OR COMPLETELY CURVILINEAR SEGMENTS

In the case of curved segments, the procedure is as follows: the start and end points of the curve are used as reference points together with the maximum curvature point. However, the qualitative description is only associated with the maximum curvature point, the other two points being used as auxiliary points for RS positioning (Figure 7). These 3 points, again, are numbered in clockwise order.

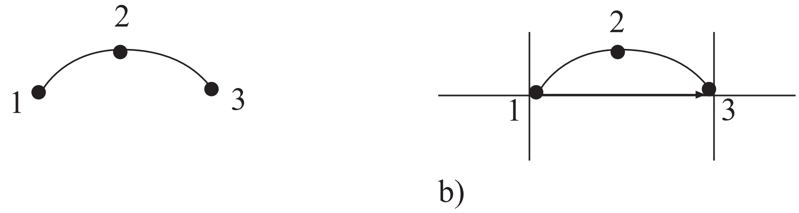


Figure 7. Example of the reference points for curved segments, and b) RS arrangement.

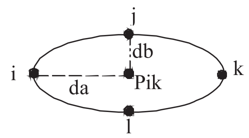
In order to determine the convexity of point j (Cj) which belongs to a curved segment, the orientation of RS is determined in the same way as in the case of the straight line segments: if reference point j remains on the left dichotomy created by the line oriented from i to k, then point j is a convex vertex. However, if point j remains on the right dichotomy created by the line oriented from i to k, then point j is concave.

The type of curvature (TCj) of point j is determined by the calculation of two distances and their comparison. In order to calculate both distances, the centre point of the line between i and k, termed point ik (Pik), is calculated first. The first calculated distance (da) is the distance between i and the new point, Pik, and the second distance (db) considered is the distance between point j and point Pik. The type of curvature is then determined by comparing both distances as follows:

If da<db → TCj = acute

If da=db → TCj = semicircle

If da>db → TCj = flat



3.3 DESCRIPTION OF OBJECTS WITH HOLES

In order to describe objects with holes, the *Completely Inside Inverse (CIi)* [13] topological concept is used, since a hole is always inside the boundaries of closed objects. The relation *CIi* is defined as the inverse of *CI* (Completely Inside) relation, which is defined in turn from the formal definition of the *In* relation as follows:

$$(h_1, \text{in}, h_2) \leftrightarrow h_1 \cap h_2 = h_1 \wedge h^{\circ}{}_1 \cap h^{\circ}{}_2 \neq \varnothing$$

If (h_1, in, h_2) is obeyed, the following algorithm distinguishes between three possible relations: completely-inside, touching-from-inside, and equal:

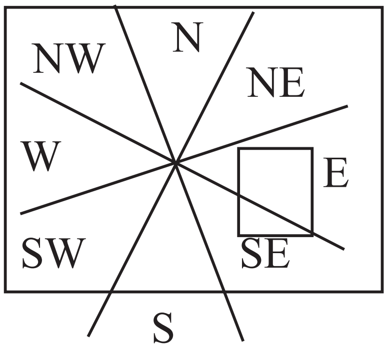
If (h_2, in, h_1) then (h_1, equal, h_2)

But if $h_1 \cap \delta h_2 \neq \varnothing$ then $(h_1, \text{touching-from-inside}, h_2)$

Otherwise $(h_1, \text{completely-inside}, h_2)$.

In order to relate each hole to its container, hole orientation is also determined with respect to the container. This is done using the reference system of cardinal directions (CRS) developed by Frank [14]. The orientation is determined positioning the origin of Frank's CRS on the centroide of the object to be described, orienting the

NW orientation towards the first vertex considered in the description of the object's boundary. Once the CRS has been positioned, the orientation of the hole with respect to the container is calculated as the set of orientations that intersect with the hole boundary. Figure 8 shows an example, in which the orientation of the hole with respect



to the container will turn out to be [E,SE]. When all the orientations occur, a special orientation termed Centre (C) is found.

Figure 8. Example of the determination of hole orientation with respect to the container.

3.4. COMPLETE DESCRIPTION OF AN OBJECT

An object being given, its complete description is defined by the following tuple: [type_hole, type_curve, [Colour, [A1, C1, L1 | Curve, C1, TC1] … [An, Cn, Ln | Curve, Cn, TCn]], CIi, Orientation, [type_curve, [AH1, CH1, LH1 | Curve, CH1, TCH1] … [AHj, CHj, LHj | Curve, CHj, TCHj]], where *n* is the number of vertices (reference points) of the container and *j* is the number of vertices (reference points) of the hole. The *type_hole* symbol may have one of the following values *[without holes, with holes]*, and the symbol *type_curve* may adopt one of the following values *[without curves, with curves, only curves]*; both symbols are introduced to accelerate the correspondence or recognition process. Colour is the RGB colour of the figure described by the three basic components, the Red, Green, and Blue coordinates. Each vector, [Ai, Ci, Li | Curve, Ci, TCi], represents a qualitative description node, which may be the description of a vertex of a straight line segment, then taking on the form [Ai,Ci,Li], with *i=1,..,n,* where A1, …, An, are the qualitative angles of each vertex; C1, …, Cn are the types of convexity of each vertex and L1, …, Ln are the relative lengths of each pair of adjacent edges to each of the vertices of the straight line segments of the container; or second tuple [Curve, Ci, TCi] which represents the qualitative description of a curvilinear segment and is formed first by the *Curve* symbol to indicate that this is a curvilinear segment and then by the labels C1, …, Cn, and TC1, …, TCn, which are the qualitative description of the type of convexity and the type of curvature, respectively. The same occurs with the hole container: AH1.. AHj, CH1, …, CHj and LH1, …LHj, are the qualitative angle, type of convexity, and relative length of the adjacent edges of each of the vertices of the straight line segments of the hole and CH1, …, CHj, and TCH1, …, TCHj are the type of convexity and type of curvature of the curvilinear segments of the hole. *CIi* is the topological relation that relates the hole to its container, as already explained in the foregoing section. Finally, the *Orientation* is one or a set of orientation relations given by the CRS, in order to provide the relative position of the hole in regard to its container. CIi, the Orientation, and the description of the hole boundaries only appear when the object is of the type with holes and will appear as many times as is the number of holes in the figure.

Therefore, in order to describe a figure fully, it is necessary first to repeat the process of giving a qualitative description of each vertex (or reference point) in order to describe the boundary of the container, starting with the vertex numbered 1. From vertex 1, and using this and vertex 3, the qualitative representation of vertex 2 is determined. This process is repeated until the last vertex (n) has been characterised, taking into account that vertex n (the last vertex of the figure) and vertex 2 will be used as reference vertices to characterise the first vertex. Once the container has been qualitatively described, the colour is stored as a vector that contains the three RGB coordinates. Then, if the figure contains holes, the orientation relation between the container and each of the holes is calculated using the CRS, thus obtaining another vector with all the orientations that each hole occupies. Finally the qualitative description of each hole boundary is constructed, following the same process as in the container boundary, bearing in mind that the initial reference point for this description is the one closest to the initial vertex of the container description. This is because it is subsequently needed in the correspondence process in order to perform a non-cyclical comparison of the hole boundaries, it being necessary to ensure that the description always begins from the same vertex. The final vector with the characteristics of the figure is thus constructed.

The following figure displays an example of a figure with a hole, straight line and curvilinear segments, and their qualitative description.

Figure 9. Example of a black figure (RGB = 0R,0G,0B) with a hole and straight and curved segments. Qualitative description of the figure 9 = [with_holes, with_curves, [[0,0,0], [straight, convex, longer], [straight, convex, longer], [curve, convex, acute], [curve, convex, acute], [straight, convex, shorter], [straight, convex, shorter], [straight, convex, longer]], CIi, [E, SE, S, SW, W], [[straight, convex, shorter], [straight, convex, longer], [straight, convex, shorter], [straight, convex, longer]]].

3.5. CORRESPONDENCE OR RECOGNITION PROCESS

This process consists of the development of an algorithm which, given a qualitative description of each of two tesserae, determines whether or not the two figures represent the same object. In this process, one of the figures is taken as the reference figure and the other as the figure to be compared.

The first step in the correspondence process consists of the construction of the full qualitative description of the reference figure, which is done by following the steps described in the previous sections.

The qualitative description of the object to be compared is then only constructed up to the complete description of its contour or boundary (the holes are not yet described). This is because if the full description of the objects with holes that are rotated in regard to the reference figure is constructed a priori, this description would define different orientations for each hole in the piece, since it is not yet known how to locate the CRS, since this CRS must have a NW orientation including the first vertex of the boundary description. If the CRS is set on another vertex that does not match the first one of the reference figure, the final set of orientations will be different.

The two descriptions (character chains) are compared in order to establish if they are of the same type (without curves, with curves, only curves, with holes, or without a hole), same colour, and same size. The comparison of these two last characteristics is performed as set out below.

Object colour is stored as the RGB coordinates in their qualitative description. In the recognition process, these coordinates are qualitatively compared using the distance Delta E between two colours.

Given two colours in RGB, designated Colour1(C1)=(R1,G1,B1) and Colour2(C2)=(R2,G2,B2), the distance Delta_E is then calculated as the Euclidean distance between both colours, as follows:

$$Delta\_E(C1,C2) = ((R1-R2)^2 + (G1-G2)^2 + (B1-B2)^2)^{1/2}$$

If Delta_E is smaller than 0.2 this means that even a person trained in the field of colour recognition cannot differentiate between both colours.

As far as tesserae size is concerned, this is also qualitatively compared. The area of each tessera is calculated, and qualitatively compared as follows: if the difference between both areas is smaller than the joint size (space left when the tesserae are placed next to each other in the final design), the two tesserae are considered of equal size.

If both tesserae are of the same type, colour, and size, the qualitative description of the containers is compared (boundaries). This comparison is a cyclical comparison because the vertices of each figure will be numbered in a different way if they are in different orientations (for example, when a figure appears rotated). In addition to performing a cyclical comparison, the correspondence process will not only provide a Boolean value to establish whether or not both containers match, but it will also indicate (if both containers match) which vertex in the figure to be compared coincides with vertex number 1 in the reference figure. This vertex of the figure to be compared will be termed, hereafter, *vertex I.*

If the tesserae contain no holes, the comparison ends here; in the opposite case, if the comparison has been positive up to this point, the qualitative description of the object to be compared is then completed (the description of its holes). This is done according to the steps below:

1. The topological relation CIi is inserted in order to relate the hole to the container.
2. The orientation relation between the hole and its container is calculated, centring the CRS on the centroide and orienting the NW towards vertex I

of the figure to be compared (which coincides with the first vertex of the description of the reference figure boundary). The set of relations obtained is inserted in the qualitative description of the figure.

3. The qualitative description of the hole boundary is calculated in the same way as the construction of the qualitative description of the contour, numbering, however, as vertex number one for this description the hole vertex closest to *vertex I*. This is required because the subsequent comparison of the respective hole boundaries must be done in a non-cyclical manner. Otherwise, situations might occur in which the containers match cyclically, and the holes are oriented in the same way, but their inner orientation in relation to the container is not the same (Figure 10), so that the two figures do therefore not match, even though a cyclical comparison of the hole boundaries would indicate that both matched. The qualitative description of the hole is included in the description of the figure.
4. Steps 1, 2, and 3 are repeated for each hole.

Figure 10. Two different objects with matching holes in different positions.

Once the complete qualitative description of the figure to be compared has been obtained, the correspondence process alone shall determine whether both holes match. The procedure below is then to be followed:

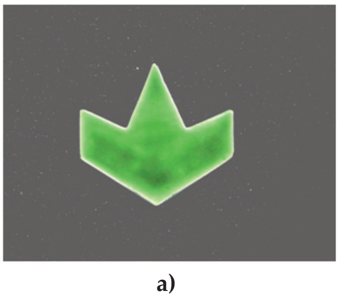
1. Take a hole from the figure to be compared.
2. Compare the orientation of this hole with that of each hole in the reference figure; if no hole with the same orientation is found, the two do not match. However, when a hole with the same orientation is found, one proceeds with the next step.
3. Make a non-cyclical comparison of the boundary of the two holes with the same orientation; if they coincide, one returns to step 1 to compare another hole. If this comparison is not solved positively, step 2 is repeated with the holes of the reference figure that have not yet been considered.

The process ends when all holes in the figure to be compared have found their counterparts in the reference figure. If there are no further holes in the reference figure, the process establishes that both figures match. In any other case, the figures do not match.

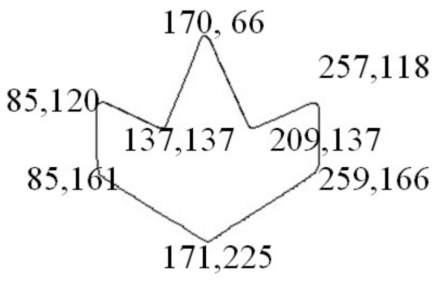
4. SOFTWARE FOR THE EXTRACTION OF RELEVANT INFORMATION FROM THE TESSERAE IMAGES

As mentioned previously, for this application it was necessary to develop an image processing algorithm in order to obtain the relevant image points. The relevant points to be extracted are the vertices, and the start, end, and maximum curvature points of the curves. This information serves to construct the qualitative description of the tesserae in the image, using the theory described in point 3. For this reason, the developed algorithm returns a file with these significant points. The developed algorithm may be classified as hybrid because it uses, first, quantitative techniques to extract the image boundary, and the result is then qualitatively processed to obtain just the points of interest. The algorithm is briefly described below; a full description may be found in [15].

The vision system uses segmentation techniques for figure contour recognition, so that a jpg image is taken by a digital camera, and the significant points of the figure extracted from this by segmentation, i.e. of their contour. The contour is obtained using the Canny method [16], considered one of the best boundary detection algorithms. The contours in the images are areas where there are strong intensity contrasts (there is a leap in intensity from one pixel to the next). Based on this criterion, the Canny algorithm detects the boundaries. For this it first smoothes the image to eliminate noise, and then calculates the image gradient in order to highlight regions with high spatial derivatives. With this image the algorithm crosses each region and eliminates non-maximum pixels. However, using the Canny algorithm still returns too many points; an additional filter has therefore been applied to the result. This filter consists of extracting the continuous line of points that form the real boundary of each object, eliminating the lines that do not form part of this boundary using the concept of the 8-neighbourhood of a pixel. From the first point resulting from the Canny algorithm, only those points are stored that are neighbours of this pixel, until the same point is reached. Thus all those points are eliminated that do not belong to the continuous line defining the object boundary, as well as all those continuous lines that do not define closed figures (Figure 8 shows an example of the result).



a)



b)

Figure 11. a) Jpg image obtained with a camera and b) the contour detected by segmentation using the Canny algorithm with the additional filter; the relevant points that were subsequently extracted using qualitative techniques have been included on this image.

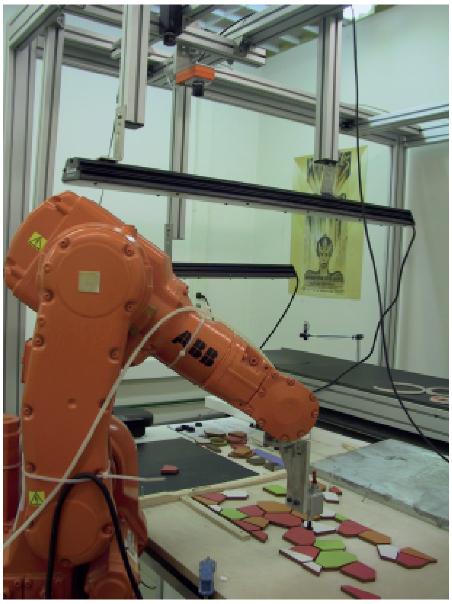


Figure 12. Image of the prototype in the laboratory

It is not of interest, however, to have to work quantitatively with all the points of the figure, even though only boundary points are involved, since if this process is conducted in a more cognitive mode, the processing time improves. Therefore, this set of points is processed qualitatively in order to determine the relevant boundary points. These points are obtained by calculating the slope between two points *Pi* and *Pj*, termed *sj*, and the slope between point *Pi* and another point *Pk*, termed *sk*. Both slopes are then compared, and the following situations may occur:

1. If sj==sk a straight segment is involved,
2. If sj≠ sk a curved segment is involved.

This process is repeated for a new point *Pl*, calculating the slope between *Pi* and *Pl (sl),* and comparing that slope with *sj*.

It should be noted that given a boundary of n points, the points selected for the calculation of the slope depend on a chosen *granularity*. For example, if the granularity is established at 20, the first point *(Pi)* will be point P_0 of the set of boundary points, Pj will be point P_{19}, Pk will be P_{39}, etc. This granularity is determined as a function of the edge length of the described object: if the edges are long, the granularity will have a larger value; if they are short, the granularity will have a smaller value.

On the other hand, the slope comparison is not exact either; a margin of error is allowed, owing to the noise found at the points.

In view of the above, to determine the relevant points, the following procedure is used:

1. If it has been determined that a straight segment is involved, the new relevant point will be the one at which the slope stops being constant.

2. If a curved segment is involved, the new relevant point is the one at which the slope changes its sign.

When this process ends (the process is repeated from each new relevant point until the first point is reached), a vector will have been obtained with the information of all the relevant image points. The information stored for each point is as follows: the absolute coordinates of each point in the image, followed by a character chain indicating whether the point belongs to a straight line or a curve. For example, for Figure 11 the vector is as follows: [(170,66), Straight line, (209,137), Straight line, (257,118), Straight line, (259,166), Straight line, (171,225), Straight line, (85,161), Straight line, (85,120), Straight line, (137,137), Straight line].

5. THE PROTOTYPE

In order to develop a prototype in the laboratory according to the specifications provided by the architecture shown in Figure 1, an IRB 140 ABB manipulating arm (an articulated arm with six axes of freedom for laboratory applications) has been used. The vision system used was a JAI M7+ –Bayer RGB LVDS camera. The camera communicates with the PC through a framegrabber card, and the IRB140 communicates with the PC through the series port.

To allow processing the camera images it was necessary to implement a Bayer filter to translate the given image in RGB Bayer into a standard RGB image.

All the described intelligent components in this paper were implemented in a PC with Windows XP as operating system (using the C++ language). It deserves to be noted that that there are two specific classes, each of which implements a controller: one for vision system control and the other for control of the manipulating arm.

The final system was implemented in the laboratory (Figure 12), its general operation being as follows: first, a mosaic design (vectorial file) is opened and the qualitative description of all tesserae in the design is created. The camera then captures images of the physical pieces on the conveyor belt, and the relevant characteristics of the tesserae in the image are extracted with subsequent construction of their description. The recognition algorithm begins, and returns which tessera in the image corresponds to which mosaic piece, as well as the rotation angle that the manipulating arm needs to travel in order to deposit this tessera in the correct orientation in the mosaic. Finally, the manipulating arm initiates its movement, picking up (with a system of suction pads) each tessera recognised by its centroide, rotating it and depositing it in the assembly area of the mosaic. This process is repeated until the mosaic is completed, or until there are no more physical pieces left to place.

6. CONCLUSIONS AND FUTURE WORK

Qualitative techniques have been applied for the independent, intelligent assembly of ceramic mosaics. The current system does not yet consider a moving conveyor belt, so that the next step will be to put that belt in motion and integrate that movement in the final system. Once this integration has been performed, operating trials will need to be run of the final system in an industrial environment, integrating this prototype with the other industrial equipment.

At present the recognition process is performed through exact recognition, but this process could be improved by developing classification techniques (using similarity distances). However, in the development of these techniques it is important to avoid false positives, since a tessera should not be incorrectly installed in the final mosaic.

7. ACKNOWLEDGEMENTS

This study has been partly funded by IMPIVA and by CICYT under project number TIN2006-14939.

REFERENCES

[1] MUSEROS, Ll. Qualitative theories on shape representation and movement: application to industrial manufacturing and robotics. Castellón: UJI, 2006. [Ph.D. thesis].

[2] LEYTON, M. A process-grammar for shape. Artif. Intel., 34, 213-247, 1988.

[3] BRADY, M. Criteria for representations of shape. In: BECK, J; HOPE, B.; ROSENFELD, A. (Eds.) Human and machine vision. New York: Academic press, 1983. p.39-84.

[4] FLYNN, P.J.; JAIN, A.K. CAD-based computer vision: from CAD models to relational graphs. IEEE Trans. P.A.M.I., 13,114-132, 1991.

[5] BIEDERMAN, I. Recognition-by-Components: A theory of human image understanding. Psychol. Rev. 94 (2), 115-147, 1987.

[6] FERRUCCI, V.; PAOLUZZI, A. Extrusion and boundary evaluation for multidimensional polyhedra. Comput. aided des., 23 (1), 40-50, 1991.

[7] JUNGERT, E. Symbolic spatial reasoning on object shapes for qualitative matching. In: Spatial Information Theory. Berlin: Springer, 1994, p. 444-462.

[8] DAMSKI, J. C.; GERO, J. S. A logic-based framework for shape representation. Comp. aided des. 28(3),169-181, 1996.

[9] COHN, A.G. A hierarchical representation of qualitative shape based on connection and convexity. In: Proceedings COSIT'95. Berlin: Springer, 1995, p. 311-326.

[10] CLEMENTINI, E.; FELICE, P. de. A global framework for qualitative shape description. GeoInformatica 1(1), 1-17, 1997.

[11] POBIL, A.P. de, SERNA, M.A. Spatial representation and motion planning. Berlin: Springer, 1995.

[12] ZIMMERMANN, K., FREKSA, C. Enhancing spatial reasoning by the concept of motion. In: Proceedings of the AISB'93. Amsterdam: IOS press, 1993. p. 140-147.

[13] ISLI, A. et al. A topological calculus for cartographic entities. Conceptualizang new regions. Lect. Notes Artfi. Intell. 1849, 225-238, 2000.

[14] FRANK, A. Qualitative spatial reasoning with cardinal directions. J. Vis. Lang. Compu. 3, 343-371, 1992.

[15] MUSEROS, Ll.; ESCRIG, M.T. Extracting relevant features of an image for qualitative shape matching. In: ORTEGA RAMÍREZ, J.A.; ABAD HERRERA, P.J.; SUÁREZ FÁBREGA, A.J. (Eds.) VII Jornadas de Trabajo ARCA, JARCA'05. Malaga: [s.n.], 2005. pp. 29-36.

[16] CANNY, J.F., A computational approach to edge detection. IEEE Trans. pattern Anal. Machin. Intel. 8, 679-714, 1986.