Joviša Žunić *

SHAPE DESCRIPTORS FOR IMAGE ANALYSIS

Abstract. We give an overview of the shape based techniques used in object matching, object identification and object classification tasks. We distinguish between the area based methods, which use all the shape points, and boundary based methods, which use boundary information only. We also discuss a recent 'multi-component shape' approach. This approach considers a group of objects as a single but compound object. The idea is already shown to be very efficient in a wide spectrum of applications.

Illustrative examples are provided, including those related to personal signature identification and outliers detections, which have, pretty much, obvious and straightforward applications in security and crime prevention related tasks.

Mathematics Subject Classification (2010): Primary: 68U10, 94A08; Secondary 62H30.

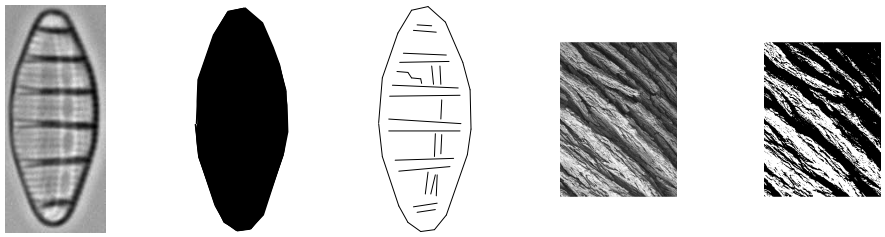
Keywords: image processing, computer vision, shape analysis, object identification, data security.

* *University of Exeter, College of Engineering, Mathematics and Physical Sciences, Exeter EX4 4QF, U. K.*
and
Mathematical Institue SANU, Kneza Mihaila 36, Belgrade, Serbia

Contents

1. Motivation and Problem Description

Image technologies have developed rapidly. A huge amount of images and image related data are available in different domains: medicine, biology, industry, geology, astronomy, crime prevention, security, etc. Different objects appear on images and they should be recognized, classified, or identified. Working in object space, i.e. compar ing object pairwise, is shown to be inaccurate and computationally expensive. It has turned out that a better idea is to map objects of interest onto a set of numbers (a vector in R^d) and then perform searching in this space (e.g., in a subset of R^d). For such mapping we need some object characteristic which can be reasonably easily and efficiently quantified by numbers. One of such characteristics can be the color of the object or its texture, for example. The shape is another object characteristic, which allows a spectrum of numerical characterizations. Also, the shape, as an object characteristics, has a big discrimination capacity. I.e., objects of different kind, very often, can be distinguished by their shapes. In Figure 1(a) an original diatom image is given. For a further processing a preprocessing is needed. In Figure 1(b) a shape of diatom is presented, while boundary shape and interior details are in Figure 1(c).

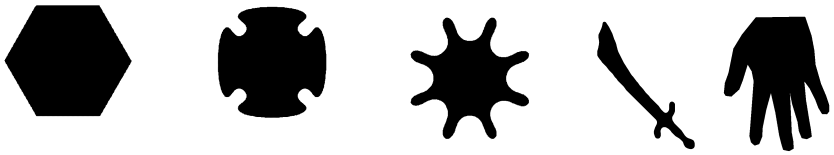


(a) (b) (c) (d) (e)

FIGURE 1. Diatom images: (a) original image; (b) whole object shape; (c) boundary shape and shape of internal contours. Texture image: (d) original image, (e) original image thresholded.

Figure 1(d) presents a texture. A corresponding black-white image is in Figure 1(e). There is no a clear shape of such extracted texture.

We illustrate the basic idea by simple shape examples in Figure 2. Shapes in Figure 2(a)-(c) are rotationally symmetric and by this property they can be distinguished from the shapes in Figure 2(d,e). The question is: *How to define the function* $\mathbf{D}_1(S)$ *which maps planar shapes into the interval* $[0, 1]$*, such that it assigns the value* 1 *to rotationally symmetric shapes, and assigns lower values to the shapes in Figure* 2(d) *(let say* 0.8*) and in Figure* 2(e) *(let say* 0.1*)?*



(a) (b) (c) (d) (e)

FIGURE 2. Shape examples.

Such a defined function $\mathbf{D}_1(S)$ could be called a 'symmetricity' measure. Further, because $\mathbf{D}_1(S)$ does not distinguish among the shapes in Figure 2(a-c), another descriptor should be considered. Since the shape in Figure 2(a) is convex, and since the shape in Figure 2(b) is 'more convex' than shape in Figure 2(c), a 'convexity' measure (i.e., another function) $\mathbf{D}_2(S)$, will do. $\mathbf{D}_2(S)$ is expected to assign 1 to convex shapes and smaller values for 'less convex' shapes (e.g., 0.92 to the shape in Figure 2(b) and 0.75 to the shape in Figure 2(c)). Such defined function, very likely, would distinguish between the shapes in Figure 2(d) and Figure 2(e) but is not clear would it distinguish among shapes in Figure 2(b) and Figure 2(d). It is difficult do judge which of them is 'more convex'. To overcome such problem, another descriptor, e.g., shape 'linearity', could be involved. A linearity measure

$\mathbf{D}_3(S)$ should assign a high value to the shape in Figure 2(d) (let say 0.9) and a small and similar values for the rest of shapes (e.g. all close to 0.1).

This is a basic idea how to use shape descriptors and corresponding measures to be able to distinguish between shapes/objects. Of course, in most cases a single descriptor and a single measure are not enough, and several of them should be combined. For example, the shape in Figure 2(e) can be separated from the others as a shape with low $\mathbf{D}_1(S)$ and $\mathbf{D}_3(S)$ values (e.g. with low both symmetricity and linearity measures).

2. Introduction

Shape descriptors are a powerful tool used in wide spectrum of computer vision and image processing tasks such as object matching, classification, recognition and identification. Many approaches have been developed [40]. There are a number of generic shape descriptors that are capable of providing a high dimensionality feature vector that accurately describes specific shapes (for example, Fourier descriptors and moment invariants). Alternatively, other descriptors describe some single characteristic that is present over a variety of shapes, such as circularity [30], ellipticity, rectangularity, triangularity [33], rectilinearity [50], complexity [29], mean curvature [21], symmetry [48], etc. Even for a single characteristic of shapes there often exist many alternative measures which are sensitive to different aspects of the shape. Very likely, the shape convexity is a shape property with the largest number of different methods defined for its evaluation–see [3, 20, 31, 36, 37, 41, 51]. The need for alternative measures is caused by the fact that there is no a single shape descriptor which is expected to perform efficiently in all possible applications.

Generally speaking, there are two approaches to analyze shapes: boundary based (which use the information from boundary points only) and area based ones (which use all the shape points). It could be said that, in the past, more attention has been given to the area based methods. The area based methods are more robust (e.g. with respect to noise). Although not mentioned often, an additional reason for a larger number of methods that are based on 'interior' shape points, rather than methods based on boundary points, is that area based methods are usually simpler to compute. For example, to estimate accurately the area of a given shape, it is sufficient to enumerate the number of pixels inside the shape [17], while the perimeter estimation is not a straightforward task. Depending on particular situation and conditions assumed different methods have to be used [7, 38].

Another example would be geometric (area) moment invariants [16]; these are easily and accurately computable from the corresponding object image, while their boundary based analogues involve computation of path integrals, which are not simple to be estimated from discrete data, which are mainly used in image processing and computer vision applications. On the other side, the boundary based methods are more suitable for a high precision computer vision and image processing tasks (person identification, for example). They are able to cope much easier with objects with partially extracted boundaries or with partially occluded objects. Robustness is a very desirable property when we work with low quality

data (e.g., noisy images or low resolution images), but recently, due to progress in image technology, high quality data can be provided, and the use of boundary based methods becomes highly acceptable in many applications. In addition, boundary based methods could have a much lower time complexity because shape boundaries are represented by a significantly smaller number of pixels than complete shapes are. Of course, there are methods which cannot be classified either as boundary based or volume (area) based ones. For example, a very popular shape measure, the shape compactness

$$\mathbf{C}_{st}(S) = \frac{4 \cdot \pi \cdot \text{Area\_of\_}S}{(\text{Perimeter\_of\_}S)^2}$$

obviously uses both boundary and interior information. This quantity indicates how much a given shape differs from a perfect circular disc, which is understood as the most compact shape. Accordingly, the highest possible compactness (equal to 1) is assigned to circular disc. Finally, there are methods which use only information from specific points (shape corners, for example) or specific boundary parts (e.g., parts belonging to the convex hull of the shape considered).

Here, we focus on shape analysis techniques based on the use of a set of suitably selected shape descriptors/measures. Generally speaking, a shape measure is a quantity which relates to a particular shape characteristic. More formally, a certain shape measure $\mathbf{D}(S)$ (related to a certain shape descriptor) maps a given planar shape S into a real number. In order to be applicable in object classification, recognition or identification task, any shape measure is expected to be invariant with respect to similarity transformations (translation, rotation, and scaling). Also, shape measures are preferred to be given in a normalized form. An easiest way to achieve a normalized form is to apply a scaling transformation which would preserve that $\mathbf{D}(S)$ varies through the interval $[0, 1]$ (or even better through the interval $(0, 1]$) while S varies through the set of bounded compact planar regions.

Thus, common desirable properties of a given shape measure $\mathbf{D}(S)$ are:

(a) $\mathbf{D}(S) \in [0, 1]$
(b) $\mathbf{D}(S) = 1$
emphif and only if S satisfies a certain property (here called a shape descriptor) for which, actually, the shape measure $\mathbf{D}(S)$ is designed.
(c) $\mathbf{D}(S)$ is invariant with respect to the similarity transformations.
(d) For any $\delta > 0$ there is a shape S such that $\mathbf{D}(S) < \delta$
(e.g., 0 is the best possible lower bound for $\mathbf{D}(S)$.)

The paper is organized as follows. In the next section we consider area based shape descriptors. Section 4 is related to boundary based shape descriptors, while Section 5 relates to the recent concept of multi-component shapes. Concluding remarks are in Section 6.

3. Area Based Shape Descriptors

As mentioned, area based shape analysis methods (including shape descriptors based approaches) are, so far, mostly studied in literature. These methods are expected to be robust (e.g., with respect to noise or with respect to narrow boundary

intrusions), and, because of that, they are very suitable when working with low quality data or with a low resolution images. In this section we discuss some of area based descriptors which are in a frequent use.

3.1. Geometric Moments and Moment Invariants. Being theoretically well founded and well understood, moments based techniques are very popular and very useful in the image processing and computer vision tasks. A number of methods were developed. We proceed with a short overview.

For a given planar shape S its geometric (area) (p,q)-moment $m_{p,q}(S)$ is defined as

$$m_{p,q}(S) = \iint_S x^p y^q dx\, dy. \tag{3.1}$$

The order of $m_{p,q}(S)$ is $p+q$. Trivially, $m_{0,0}(S)$ equals the area of S. When work in discrete space, i.e., when a real shape S is represented with its digitization $dig(S)$, then $m_{p,q}(S)$ is approximated as

$$m_{p,q}(S) = \iint_S x^p y^q dx\, dy \approx \sum_{\text{pixel } (i,\,j) \text{ belongs to dig}(S)} i^p \cdot j^q$$

if the pixel size is assumed to be 1×1.

Obviously, $\sum_{\text{pixel } (i,\,j) \text{ belongs to dig}(S)} i^p \cdot j^q$ is very simple to compute (only summations and multiplications are needed) and the approximation in (3.1) is very accurate [22]. There are also methods for a fast computation of such an approximation–e.g., [19, 24]. These are reasons why moments are used to define very common features in image processing and computer vision applications. For example, one of the basic shape features, as it is the shape position, is usually expressed in terms of moments. Precisely, the position of a given shape S is described by the shape centroid $(x_c(S), y_c(S))$ which is defined as

$$(x_c(S), y_c(S)) = \left(\frac{m_{1,0}(S)}{m_{0,0}(S)},\ \frac{m_{0,1}(S)}{m_{0,0}(S)} \right). \tag{3.2}$$

Since moments $m_{p,q}(S)$ are not translation invariant (e.g., if S moves the corresponding moments change) it is suitable to consider the central moments $\overline{m}_{p,q}(S)$ which are defined as

$$\overline{m}_{p,q}(S) = \iint_S (x - x_c(S))^p\, (y - y_c(S))^q\, dx\, dy$$

and which are translation invariant by definition.

Further, because isometric objects could appear on an image as object of different size (depending on their position with respect to the camera) it is suitable to have object features which are scaling invariant. Since $\overline{m}_{p,q}(S)$ are not scaling invariant, it is convenient to involve, so called, normalised moments. A normalized moment

$\mu_{p,q}(S)$ is defined as

$$\mu_{p,q}(S) = \frac{\overline{m}_{p,q}(S)}{m_{0,0}(S)^{1+\frac{p+q}{2}}}.$$

It is easy to verify that the normalized moments $\mu_{p,q}(S)$ do not change if a given shape S is scaled for a factor $\mathbf{r}$. In other words, if S is replaced with $\mathbf{r} \cdot S = \{(\mathbf{r} \cdot x, \mathbf{r} \cdot y) \mid (x, y) \in S\}$, then $\mu_{p,q}(S) = \mu_{p,q}(\mathbf{r} \cdot S)$.

Finally, in many object classification tasks, identical (or very similar) objects have to be grouped together. Since the objects presented on an image may be placed arbitrarily, descriptors which do not depend on the object position and object orientation are needed for such a grouping. This means that, apart from being translation and scaling invariant, we need shape descriptors which are rotationally invariant, as well. In his seminal work [16], Hu has introduced a set of, so called, algebraic invariants. These invariants are listed below:

$$\begin{aligned}
I_1 &= \mu_{2,0} + \mu_{0,2} \\
I_2 &= (\mu_{2,0} - \mu_{0,2})^2 + 4 \cdot (\mu_{1,1})^2 \\
I_3 &= (\mu_{3,0} - 3 \cdot \mu_{1,2})^2 + (3 \cdot \mu_{2,1} - \mu_{0,3})^2 \\
I_4 &= (\mu_{3,0} + \mu_{1,2})^2 + (\mu_{2,1} + \mu_{0,3})^2 \\
I_5 &= (\mu_{3,0} - 3 \cdot \mu_{1,2}) \cdot (\mu_{3,0} + \mu_{1,2}) \cdot [(\mu_{3,0} + \mu_{1,2})^2 - 3 \cdot (\mu_{2,1} + \mu_{0,3})^2] \\
&\quad + (3 \cdot \mu_{2,1} - \mu_{0,3}) \cdot (\mu_{2,1} + \mu_{0,3}) \cdot [3 \cdot (\mu_{3,0} + \mu_{1,2})^2 - (\mu_{2,1} + \mu_{0,3})^2] \\
I_6 &= (\mu_{2,0} - \mu_{0,2}) \cdot [(\mu_{3,0} + \mu_{1,2})^2 - (\mu_{2,1} + \mu_{0,3})^2] \\
&\quad + 4 \cdot \mu_{1,1} \cdot (\mu_{3,0} + \mu_{1,2}) \cdot (\mu_{2,1} + \mu_{0,3}) \\
I_7 &= (3 \cdot \mu_{2,1} - \mu_{0,3}) \cdot (\mu_{3,0} + \mu_{1,2}) \cdot [(\mu_{3,0} + \mu_{1,2})^2 - 3 \cdot (\mu_{2,1} + \mu_{0,3})^2] \\
&\quad + (\mu_{3,0} - 3 \cdot \mu_{1,2}) \cdot (\mu_{2,1} + \mu_{0,3}) \cdot [3 \cdot (\mu_{3,0} + \mu_{1,2})^2 - (\mu_{2,1} + \mu_{0,3})^2].
\end{aligned}$$

Because the normalized moments $\mu_{p,q}$ were used, the quantities $I_1, I_2, \ldots, I_7$ are translation and scaling invariant by definition. It also can be verified that $I_1, I_2, \ldots, I_7$ are invariant with respect to rotations. For more details and recent developments we refer the reader to [12, 13].

3.2. Shape Orientation. Geometric moments are also used to determine the shape orientation, which is, together with shape position (usually defined by the shape centroid (3.2)), a necessary part of an image normalization procedure. The most standard method for the computation of the shape orientation is based on the, so called, axis of the least second moment of inertia [18, 40]. The axis of the least second moment of inertia is the line which minimizes the integral of the squared distances of the points (belonging to the shape) to the line. The integral which should be minimized is

$$I(\alpha, S, \rho) = \iint_S r^2(x, y, \alpha, \rho)\, dx\, dy$$

where $r(x, y, \alpha, \rho)$ is the perpendicular distance from the point $(x, y) \in S$ to the line given in the form

$$X \cdot \sin\alpha - Y \cdot \cos\alpha = \rho.$$

It is easy to check the axis of the least second moment of inertia passes through the shape centroid $(x_c(S), y_c(S))$ (see (3.2)). So, if the shape S' is the translation of S by the vector

$$-\overrightarrow{\left(\frac{m_{1,0}(S)}{m_{0,0}(S)}, \frac{m_{0,1}(S)}{m_{0,0}(S)}\right)} = -\overrightarrow{(x_c(S), y_c(S))}$$

then the centroid of S' coincides with the origin. This allows us to set $\rho = 0$ and proceed with the minimization of $I(\alpha, S', \rho = 0)$ instead of the minimization of $I(\alpha, S, \rho)$.

The squared distance $r^2(x, y, \alpha, \rho = 0)$ of a point (x, y) to the line $X \cdot \sin\alpha - Y \cdot \cos\alpha = 0$ is

$$(x \cdot \sin\alpha - y \cdot \cos\alpha)^2,$$

and, if for a shorten notation $F(\alpha, S) = I(\alpha, S', \rho = 0)$, the minimizing function can be expressed as follows

$$\begin{aligned} F(\alpha, S) &= \iint_S \big((x - x_c(S)) \cdot \sin\alpha - (y - y_c(S)) \cdot \cos\alpha\big)^2 dx\, dy \\ &= \sin^2\alpha \cdot \iint_S (x - x_c(S))^2\, dx\, dy + \cos^2\alpha \cdot \iint_S (y - y_c(S))^2 dx\, dy \\ &\quad - \sin(2\alpha) \cdot \iint_S (x - x_c(S))\,(y - y_c(S))\, dx\, dy \\ &= \sin^2\alpha \cdot \overline{m}_{2,0}(S) + \cos^2\alpha \cdot \overline{m}_{0,2}(S) - \sin(2\alpha) \cdot \overline{m}_{1,1}(S). \end{aligned} \tag{3.3}$$

The angle α for which the function $F(\alpha, S)$ (i.e., the integrals $I(\alpha, S', \rho = 0)$ and $I(\alpha, S, \rho)$) reaches its minimum defines the orientation of the shape S. We give a formal definition.

Definition 3.1. The orientation of a given shape S is determined by the angle α where the function $F(\alpha, S)$ reaches its minimum.

Shape orientation, as given by Definition 3.1, is easy to compute and can be expressed in terms of moments. Indeed, since the points (angles) where $F(\alpha, S)$ riches its maxima and minima are angles (points) where the first derivative $dF(\alpha, S)/d\alpha$ vanishes, i.e., where

$$\frac{dF(\alpha, S)}{d\alpha} = \overline{m}_{2,0}(S) \cdot \sin(2\alpha) - \overline{m}_{0,2}(S) \cdot \sin(2\alpha) - 2\overline{m}_{1,1}(S) \cdot \cos(2\alpha) = 0$$

we obtain immediately that the angle α which defines the orientation of S satisfies the following equation:

$$\frac{\sin(2\alpha)}{\cos(2\alpha)} = \frac{2 \cdot \overline{m}_{1,1}(S)}{\overline{m}_{2,0}(S) - \overline{m}_{0,2}(S)}. \tag{3.4}$$

Although the standard method is naturally defined, straightforward and efficient to compute it breaks down in some circumstances. For example, problems arise when working with symmetric shapes [45, 49], but the method does not tell what the orientation should be even for some irregular shapes. If we consider the equality (3.3) we can see easily that $F(\alpha, S)$ becomes a constant function if

$$\overline{m}_{2,0}(S) - \overline{m}_{0,2}(S) = 0 \quad \text{and} \quad \overline{m}_{1,1}(S) = 0. \tag{3.5}$$

Naturally, if $F(\alpha, S)$ is a constant function (for a given shape S), none of directions α could be pointed out as the shape orientation, and thus, the standard method fails.

This has caused the development of other methods, e.g., [6, 14, 18, 39, 45] and many more, for the computation of the shape orientation. Suitability of those methods strongly depends on particular application. It is not possible to say which of them is the best one or to establish a strict ranking among them as they each have their relative strengths and weaknesses (e.g., relating to robustness to noise, classes of shape that can be oriented, computational efficiency). A method dominant at one of applications could fail at another.

Notice that difficulties in the computation of the shape orientation can be caused by the nature of certain shapes. While for many shapes their orientations are intuitively clear and can be computed relatively easily, the orientation of some other shapes may be ambiguous or ill defined. Problems related to the estimation of the degree to which a shape has a distinct orientation are considered in [53].

3.3. Shape Elongation. Observations related to the computation of the shape orientation by the axis of the second least moment of inertia, easily lead to the definition of a new shape descriptor, named the shape elongation. It is naturally to predict that a given shape is said to be elongated (in a natural meaning of the word 'elongation') if it has a distinct orientation. I.e., the minima and maxima of the optimizing integral $I(S, \alpha, \rho)$ should differ essentially for more elongated shapes. Both, minima and maxima of $F(\alpha, S)$ are easy to compute. The minimum of the integral $I(S, \alpha, \rho)$ (also the minimum of $F(\alpha, S)$) is

$$\min_{\substack{\rho \geqslant 0 \\ \alpha \in [0, 2\pi]}} I(S, \alpha, \rho) = \frac{\overline{m}_{2,0}(S) + \overline{m}_{0,2}(S) - \sqrt{4 \cdot (\overline{m}_{1,1}(S))^2 + (\overline{m}_{2,0}(S) - \overline{m}_{0,2}(S))^2}}{2}$$

and is reached for $\rho = 0$ and α satisfying (3.4). This is in accordance with the fact that the axis of least second moment of inertia passes through the origin.

The maximum of the integral $I(S, \alpha, \rho)$, if $\rho \neq 0$ is allowed, obviously does not exist (i.e., the maximum is ∞). However, if $\rho = 0$ is assumed then

$$\max_{\substack{\rho = 0 \\ \alpha \in [0, 2\pi]}} I(S, \alpha, \rho) = \frac{\overline{m}_{2,0}(S) + \overline{m}_{0,2}(S) + \sqrt{4 \cdot (\overline{m}_{1,1}(S))^2 + (\overline{m}_{2,0}(S) - \overline{m}_{0,2}(S))^2}}{2}.$$

Now, we define the ratio between

$$\max_{\substack{\rho = 0 \\ \alpha \in [0, 2\pi]}} I(S, \alpha, \rho) = \max_{\alpha \in [0, \pi)} F(\alpha, S) \quad \text{and} \quad \min_{\substack{\rho \geqslant 0 \\ \alpha \in [0, 2\pi]}} I(S, \alpha, \rho) = \min_{\alpha \in [0, \pi)} F(\alpha, S)$$

as a measure for the elongation of S.

Definition 3.2. Let a given shape S. Then the elongation $\mathbf{E}_{st}(S)$ of S is defined as

$$\mathbf{E}_{st}(S) = \frac{\overline{m}_{2,0}(S) + \overline{m}_{0,2}(S) + \sqrt{4 \cdot (\overline{m}_{1,1}(S))^2 + (\overline{m}_{2,0}(S) - \overline{m}_{0,2}(S))^2}}{\overline{m}_{2,0}(S) + \overline{m}_{0,2}(S) - \sqrt{4 \cdot (\overline{m}_{1,1}(S))^2 + (\overline{m}_{2,0}(S) - \overline{m}_{0,2}(S))^2}}. \tag{3.6}$$

Shape elongation measure, as defined by the Definition 3.6, has several desirable properties. It reaches the minimal possible value of 1 for a circle. This matches our perception that a circle has the lowest possible elongation. Also, if we consider the rectangle $R(t)$ whose edge lengths are 1 and t, then the elongation $\mathbf{E}_{st}(R(t))$ tends to ∞, as $t \to \infty$. This is also in accordance with our perception. Let us mention that $\mathbf{E}_{st}(S)$ varies through $[1, \infty)$ and from the traditional reason is not normalized to be ranging in the interval $[0, 1]$, as preferred and mentioned in the introduction. A disadvantage of the standard elongation measure is that all the rotationally symmetric shapes but also some irregular shapes, have the elongation $\mathbf{E}_{st}(S)$ equal to 1. Those shape satisfy the conditions given in (3.5). In order to avoid such problems, some generalization of $\mathbf{E}_{st}(S)$ are suggested in [49].

Let us mention that there are also some naive methods to measure the shape elongation. For example, the shape elongation can be measured as the ratio of the edges of the minimum area rectangle which encloses the measured shape. It is worth mentioning that such bounding rectangles are easy to compute [10].

3.4. Shape Circularity. Hu invariants were introduced almost 50 years ago [16]. Many related aspects have been investigated, but there is still an ongoing interest. Recently, [47] considers geometric moment invariants and shows that the Hu invariants are particular case of geometric invariants. A new related problem was, first time, considered in another recent paper [56], where the authors consider shapes which optimize certain invariants. The following theorem, which shows that the first Hu invariant $I_1 = \mu_{2,0}(S) + \mu_{0,2}(S)$ is optimized by a circle, has been proved.

Theorem 3.1. *Let S be a given planar compact shape. Then* (a) $I_1(S) \geqslant \frac{1}{2\pi}$ *and* (b) $I_1(S) = \frac{1}{2\pi} \Leftrightarrow S$ *is a circle.*

Such a nice result, which says that $I_1(S) = \mu_{2,0}(S) + \mu_{0,2}(S)$ reaches its minimum $1/(2\pi)$ if and only if S is a circle, suggests the following definition of a shape circularity measure.

Definition 3.3. The circularity measure $\mathbf{C}(S)$, of a given shape S, is defined as

$$\mathbf{C}(S) = \frac{1}{2\pi \cdot (\mu_{2,0}(S) + \mu_{0,2}(S))} = \frac{1}{2\pi \cdot I_1(S)}.$$

The circularity measure $\mathbf{C}(S)$, defined as above, has several desirable properties, as summarized in the following theorem.

Theorem 3.2. *The circularity measure $\mathbf{C}(S)$ satisfies:*

(a) $\mathbf{C}(S) \in (0, 1]$, *for all shapes S.*
(b) $\mathbf{C}(S) = 1 \Leftrightarrow S$ *is a circle.*

(c) $\mathbf{C}(S)$ *is an invariant w.r.t. similarity transformations.*
(d) *For each* $\delta > 0$ *there is a shape* S *such that* $0 < \mathbf{C}(S) < \delta$.

Of course, the circularity, as one basic shape characteristics/descriptors, has already been considered in the literature. There are several measures. The most standard one considers the relation between the shape area and the shape perimeter [40]. Exploiting the fact that the circle has the largest area among all the shapes with the same perimeter, the most standard method defines the shape circularity $\mathbf{C}_{st}(S)$ in the following way

$$\mathbf{C}_{st}(S) = \frac{4 \cdot \pi \cdot \text{Area\_of\_}S}{(\text{Perimeter\_of\_}S)^2}. \tag{3.7}$$

The measure $\mathbf{C}_{st}(S)$ satisfies the properties (a)–(d) listed in Theorem 3.2. The proof is easy and straightforward. Notice that $\mathbf{C}_{st}(S)$ cannot be classified neither as area based nor boundary based because it uses both interior points (the shape area is needed) and boundary points (the shape perimeter is used for the computation).

Now we give several examples to illustrate the behavior of these two, $\mathbf{C}(S)$ and $\mathbf{C}_{st}(S)$, circularity measures.

The first example is in Figure 3. Ten fish shapes are ranked with respect to their measured $\mathbf{C}(S)$ circularity (the numbers given immediately below the shapes). The obtained ranking $(a)(b)(c)(d)(e)(f)(g)(h)(i)(j)$ is pretty much in accordance with our perception.

If the same shapes are ranked with respect to $\mathbf{C}_{st}(S)$, a different ranking $(b)(a)(c)(d)(e)(g)(h)(f)(i)(j)$ is obtained. Such a different ranking is expected, but also preferred, because the different rankings obtained suggest that a use of both measures could increase the classification efficiency. The standard circularity measure $\mathbf{C}_{st}(S)$ penalizes deep intrusions into the shape, because such intrusions lead to an essential perimeter increase. Consequently, deep intrusions imply a lower $\mathbf{C}_{st}(S)$ circularity. The measure $\mathbf{C}(S)$ is area based and does not penalizes such intrusions. This explains why the shape in Figure 3(a) has a higher measured $\mathbf{C}(S)$ circularity than the shape in Figure 3(b). On the other hand, the measure $\mathbf{C}_{st}(S)$ penalizes intrusions into the shape in Figure 3(a) and assigns a higher measured circularity $\mathbf{C}_{st}(S)$ to the shape in Figure 3(b). The eight position of the shape in Figure 3(f), if ranked by $\mathbf{C}_{st}(S)$, can be explained on a similar way.

The second example is in Figure 4. The selected shapes illustrate the robustness of $\mathbf{C}(S)$ and the sensitivity $\mathbf{C}_{st}(S)$. All four presented shapes have very similar $\mathbf{C}(S)$ circularity even though the fourth shape (in Figure 4(d)) has a very high noise level. Such obtained measures are caused by the fact that $\mathbf{C}(S)$ is an area based measure and, because of that, is very robust. On the other hand, $\mathbf{C}_{st}(S)$ can only cope with small levels of noise because it uses the shape perimeter for the computation. Indeed, the shape in Figure 4(a) has more than 2 times higher $\mathbf{C}_{st}(S)$ circularity than the shape in Figure 4(d).

The third example is in Figure 5. A big advantage of $\mathbf{C}(S)$ over $\mathbf{C}_{st}(S)$ is demonstrated by using simple synthetic shapes. All three (compound) shapes displayed consist of three isometric circular discs–see Figure 5(a)–(c). In all three cases the same the standard circularity measure $\mathbf{C}_{st}(S)$ (equal to 1/3) is assigned. This is

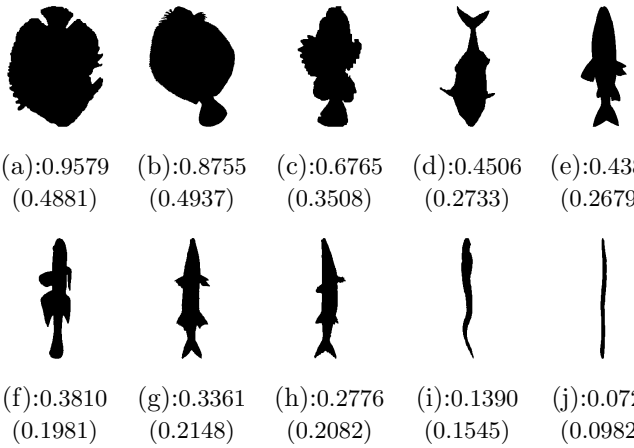
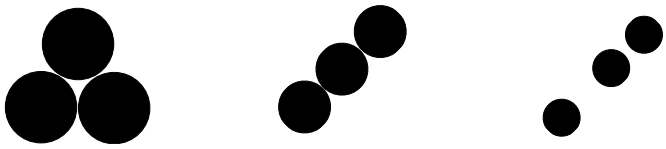


FIGURE 3. Fish shapes are ranked with respect to their $\mathbf{C}(S)$ circularities (numbers immediately below the shapes). $\mathbf{C}_{st}(S)$ values are in brackets.

FIGURE 4. $\mathbf{C}(S)$ circularities of shapes with added noise are given immediately below the shapes. The corresponding $\mathbf{C}_{st}(S)$ circularities are in brackets.

in accordance with the definition, see (3.7), because all three compound shapes have the same area and the same perimeter (the sum of perimeters of the shape components). On the other hand, $\mathbf{C}(S)$ assigns different circularities $\mathbf{C}(S)$ to the shapes in Figure 5(a)–(c). The computed circularities $\mathbf{C}(S)$ depend on the mutual position of the discs inside the shape, what is our preference.

3.5. Family of Circularity Measures. The method used to define circularity measure $\mathbf{C}(S)$ allows an extension to a family of circularity measures [56]. It has been shown that the measures from the family behave differently, implying that some of them can be combined in order to increase the classification performance. The key statements used for the extension of the $\mathbf{C}(S)$ measure to a family of circularity measures are given by the following two lemmas. Proofs can be found in [56].



(a): 0.7909 (b): 0.4540 (c): 0.2091

FIGURE 5. Measured circularity $\mathbf{C}(S)$ of a compound shape S depends on the mutual position of the components of S. All three compound shapes have $\mathbf{C}_{st}(S)$ circularity equal to 1/3.

Lemma 3.1. *Let S be a planar compact shape whose centroid coincides with the origin, and let a constant $\beta > 0$. Then the following statements hold*

$$\frac{1}{(\mu_{0,0}(S))^{\beta+1}} \iint_S (x^2+y^2)^\beta dx\,dy \geqslant \frac{1}{\pi^\beta \cdot (\beta+1)}$$

$$\frac{1}{(\mu_{0,0}(S))^{\beta+1}} \iint_S (x^2+y^2)^\beta dx\,dy = \frac{1}{\pi^\beta(\beta+1)} \;\Leftrightarrow\; S \text{ is a circle.}$$

Lemma 3.2. *Let S be a planar compact shape whose centroid coincides with the origin and let β be a constant $-1 < \beta < 0$. Then*

$$\frac{1}{(\mu_{0,0}(S))^{\beta+1}} \iint_S (x^2+y^2)^\beta dx\,dy \leqslant \frac{1}{\pi^\beta \cdot (\beta+1)}$$

$$\frac{1}{(\mu_{0,0}(S))^{\beta+1}} \iint_S (x^2+y^2)^\beta dx\,dy = \frac{1}{(\beta+1)\pi^\beta} \;\Leftrightarrow\; S \text{ is a circle.}$$

The following definition comes naturally from the arguments from the previous two lemmas.

Definition 3.4. Let S be a shape whose centroid coincides with the origin and let a real β such that $-1 < \beta$ and $\beta \neq 0$. Then the circularity measure $\mathbf{C}_\beta(S)$ is defined as

$$\mathbf{C}_\beta(S) = \begin{cases} \dfrac{\mu_{0,0}(S)^{\beta+1}}{(\beta+1)\cdot\pi^\beta \cdot \iint_S (x^2+y^2)^\beta dx\,dy}, & \beta > 0 \\ \dfrac{(\beta+1)\cdot\pi^\beta \cdot \iint_S (x^2+y^2)^\beta dx\,dy}{\mu_{0,0}(S)^{\beta+1}}, & \beta \in (-1,0). \end{cases}$$

It is worth mentioning that the measures $\mathbf{C}_\beta(S)$, $\beta \in (-1,0)\cup(0,\infty)$, satisfy the following properties:

(a) $\mathbf{C}_\beta(S) \in (0,1]$ for all planar shapes S.
(b) $\mathbf{C}_\beta(S) = 1 \;\Leftrightarrow\; S$ is a circle.
(c) $\mathbf{C}_\beta(S)$ is invariant with respect to similarity transformations.
(d) For each $\delta > 0$ there is a shape S such that $0 < \mathbf{C}_\beta(S) < \delta$.

For proof details we refer to [56].

Now we give some examples. More details are in [56]. Circularities were measured for the set of 54 masses from mammograms, combining images from the MIAS and Screen Test databases [32], see Figure 6. Rangayyan et al. [32] assessed the measures by classifying them as circumscribed/spiculated, benign/malignant, and CB/CM/SB/SM, in two group and four group classification experiments. Their best shape measure results for the three classification tasks were:

1. Circumscribed versus spiculated: 88.9% achieved by both $\mathbf{C}_{st}(S)$ and a Fourier based shape factor.
2. Benign versus malignant: 75.9% achieved by the Fourier based shape factor.
3. Four-way discrimination: 64.8% achieved by both $\mathbf{C}_{st}(S)$ and the Fourier based shape factor.

From Table 1 we see that the best results from using $\mathbf{C}_{\beta}(S)$ occurred for $\beta = 32$ and were respectively better, worse, and equal to Rangayyan et al.'s. Circularity measures from [15, 30] did not perform as well as $\mathbf{C}_{\beta}(S)$.

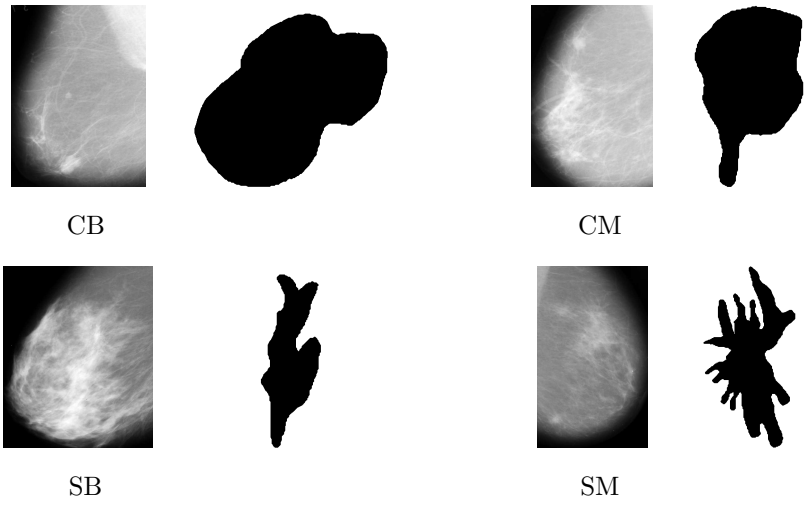


FIGURE 6. Examples of the four classes of mammographic masses: circumscribed benign (CB), circumscribed malignant (CM), spiculated benign (SB), spiculated malignant (SM). The masses were extracted from the mammograms on the left, and have been drawn rescaled.

4. Boundary Based Shape Descriptors

Boundary based methods become more popular in the recent days. That is caused mainly by a strong demand for a higher precision in image processing and

circularity measure	mammography		
	circ./spic.	mal./ben.	4 groups
$\mathbf{C}_{\beta=1/8}(S)$	83.33	66.67	51.85
$\mathbf{C}_{\beta=1/4}(S)$	85.19	64.81	51.85
$\mathbf{C}_{\beta=1/2}(S)$	75.93	57.41	42.59
$\mathbf{C}_{\beta=1}(S)$	68.52	68.52	51.85
$\mathbf{C}_{\beta=2}(S)$	75.93	68.52	53.70
$\mathbf{C}_{\beta=4}(S)$	72.22	46.30	33.33
$\mathbf{C}_{\beta=8}(S)$	79.63	59.26	50.00
$\mathbf{C}_{\beta=16}(S)$	87.04	57.41	51.85
$\mathbf{C}_{\beta=32}(S)$	**90.74**	**70.37**	**64.81**
$\mathbf{C}_{st}(S)$ pix.	87.04	59.26	57.41
$\mathbf{C}_{st}(S)$ pol.	85.19	59.26	57.41
Haralick [15]	68.52	46.30	37.04
Proffitt [30]	51.85	42.59	25.93

TABLE 1. Applications of the circularity measures to classification of mammographic masses. The second, third and fourth columns report classification accuracies. Results for the best performing measure for each task is highlighted in bold.

computer vision tasks. Another reason is that, due to the permanent development in the image technology, a higher quality data can be provided. Despite boundary based approaches are less robust and very often theoretically more complicated, they have some obvious advantages. Apart from a higher precision, it is worth mentioning that boundary based methods can cope with particularly extracted boundaries and with objects which are linear in their nature (signatures, for example). Obviously, the later objects cannot be treated by area based methods. In addition, boundary based methods are usually faster to compute (the boundary consists a smaller number of pixels than the whole shape does).

Notice that sometimes there is an easy (at least theoretical) extension of the area based methods to their boundary analogues, or vice-versa. There are also situations where this is not a simple task. An example could be the rectilinearity measure [50] used to detect buildings on satellite images, whose area based analogue is not discovered yet.

4.1. Line Moments. There is an obvious analogue for geometric (area) moments introduced by (3.1). If a curve γ is given in an arc-length parametrization,

$$\gamma: \quad x = x(s), \;\; y = y(s), \;\; s \in [0, \tau]$$

then the line moment $\eta_{p,q}(\gamma)$ is defined as $\eta_{p,q}(\gamma) = \int_\gamma x(s)^p y(s)^q ds$. Obviously, $\eta_{0,0}(\gamma) = \int_\gamma ds = \tau$ equals the length of the curve γ. Of course, γ can be the boundary ∂S of a bounded planar region S, but also it can be an open curve.

Central moments $\overline{\eta}_{p,q}(\gamma)$ are defined as

$$\overline{\eta}_{p,q}(\gamma) = \int_\gamma \left(x(s) - \frac{\eta_{1,0}(S)}{\tau} \right)^p \cdot \left(y(s) - \frac{\eta_{0,1}(S)}{\tau} \right)^q ds,$$

while the normalized moments $\zeta_{p,q}(\gamma)$ are defined as

$$\zeta_{p,q}(\gamma) = \frac{\overline{\eta}_{p,q}(\gamma)}{(\eta_{0,0}(\gamma))^{1+p+q}}.$$

Many of statements and methods based on a use of geometric (area) moments have a straightforward extension to analogue statements based on a use of line moments. Such an example are Hu invariants $I_1, I_2, \ldots, I_7$ listed in Section 3.1. It is enough to replace the integrals $\iint_S x^p y^q dx\, dy$, appearing in $I_1, I_2, \ldots, I_7$, with their analogue path/line integrals $\int_{\partial S} x(s)^p y(s)^q ds$, (where the boundary ∂S is given in an arc-length parametrization form: $x = x(s)$, $y = y(s)$) and all seven invariants remain valid. Notice that it is crucial that the boundary ∂S is given in an arc-length parametrized form in order to preserve the invariance.

4.2. Boundary based shape orientation. The standard method for the computation of the shape orientation, based on the axis of the least second moment of inertia, has its boundary based analogue. Informally speaking, we can define the shape orientation by the line which minimizes the line integral of the squared distance of the boundary points to the line, or more formally by the line which minimizes the integral

$$I(\alpha, \partial S, \rho) = \int_{\partial S} p^2(x, y, \alpha, \rho)\, ds \tag{4.1}$$

where $p^2(x, y, \alpha, \rho)$ is the distance from the point $(x, y) \in \partial S$ to the line given in the form $X \cdot \sin\alpha - Y \cdot \cos\alpha = \rho$. The boundary ∂S has to be is given in an arc-length parametrization. Following the same formalism, as in the case of the the standard method, we can deduce that the angle α which defines such a defined shape orientation satisfies the following equation

$$\frac{\sin(2\alpha)}{\cos(2\alpha)} = \frac{2 \cdot \overline{\eta}_{1,1}(\partial S)}{\overline{\eta}_{2,0}(\partial S) - \overline{\eta}_{0,2}(\partial S)}. \tag{4.2}$$

Exploiting boundary points, to define the shape orientation, gives additional possibilities for new shape orientation methods. Some specific boundary features can be involved. Notice that boundary details may not play any essential role when work with area based methods. That is because changes in boundary details could lead to very small changes in the area of shape and in related features.

The recent paper [27] defines the shape orientation by the angle α which maximizes the integral of the weighted squared lengths $|\mathbf{pr}_\alpha(\overrightarrow{x'(s), y'(s)})|^2$ of the projections $\mathbf{pr}_\alpha(\overrightarrow{x'(s), y'(s)})$ of all the tangent vectors $(\overrightarrow{x'(s), y'(s)})$, to the shape boundary ∂S, onto a line having the slope α (see Figure 7). The definition is natural and well motivated. The weights, dependent on the curvature at the boundary points, allow us to tune the method behavior. For example, by a suitable choice of the weights,

it is possible to give a higher impact to the straight sections on the boundary, or to the sections with a high curvature. A formal definition follows.

Definition 4.1. Let a shape S be given. Let $x = x(s)$, $y = y(s)$, $s \in [0,1]$ be an arc-length parametrization of the boundary ∂S. Also, let $f(\kappa(s))$ be a function dependent on $\kappa(s)$ which is the curvature of ∂S at the point $(x(s), y(s))$. The orientation $O_f(S)$ of the shape S is defined by the slope α that maximizes the integral

$$I_f(\alpha, \partial S) = \int_{\partial S} f(\kappa(s)) \cdot |\mathbf{pr}_\alpha \overrightarrow{(x'(s), y'(s))}|^2 ds$$

of the squared lengths of the projections of all the tangent vectors of ∂S on this line, weighted by the curvature at the boundary ∂S points.

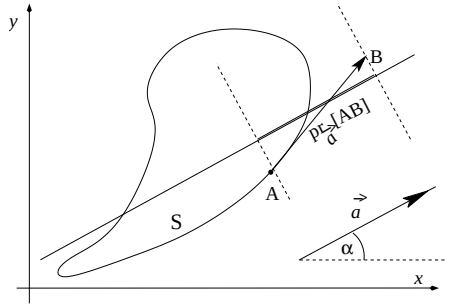


FIGURE 7. Projection of a tangent vector $(x'(s), y'(s)) = \vec{AB}$ onto a line having the slope α.

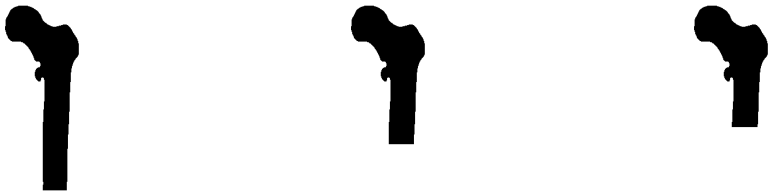
An obvious advantage of the new measure is that it has tuning possibilities. Also, Definition 4.1 enables a closed formula for the computation of the shape orientation. This is shown by the following theorem (for a proof see [27]).

Theorem 4.1. *Assume a given shape S and a function $f(\kappa)$. Then the orientation $O_f(S)$ satisfies*

$$\frac{\sin(2O_f(S))}{\cos(2O_f(S))} = \frac{2\int_{\partial S} f(\kappa(s))\, x'(s)\, y'(s)\, ds}{\int_{\partial S} f(\kappa(s))(x'(s)^2 - y'(s)^2)\, ds} \tag{4.3}$$

where $x = x(s)$, $y = y(s)$, $s \in [0,1]$ is an arc-length parametrization of the boundary ∂S and $\kappa(s)$ is the curvature of ∂S at the point $(x(s), y(s))$.

Examples in Figure 8 illustrate how a suitable choice of the weighting function could lead to the preferred method behavior. The same bone has been framed differently in each sub-picture. The orientations computed by the standard method (see (3.4)) gives inconsistent orientations: 107.0°, 120.4° and 131.0°. Let us notice that due to the nature of the object presented, the vertical orientation or, at least, a nearly vertical orientation is preferred. Equally weighting every boundary point ($f(\kappa(s)) = 1$) still produces rather different orientations for each frame (91.7°, 95.0° and 97.4°).



(a) (b) (c)

	standard method	$f(\kappa(s)) = 1$	$f(\kappa(s)) = e^{-0.01\cdot\kappa(s)}$	$f(\kappa(s)) = e^{-0.05\cdot\kappa(s)}$
(a)	107.0°	91.7°	91.7°	91.7°
(b)	120.4°	95.0°	95.0°	91.7°
(c)	131.0°	97.4°	97.4°	91.7°

FIGURE 8. The same bone is captured in different frames and orientations of each frame are computed by the standard method and by using different weighting functions in (4.3).

However by using a weighting function $f(\kappa(s)) = e^{-0.05\cdot\kappa(s)}$, higher weights are given to boundary points with a small curvature (i.e., to the points on the straight sections of the boundary), and the same orientation of 91.7° is computed for each frame. In addition, the computed orientations are nearly vertical, as preferred.

The following discussion points out advantages of the measure given by Definition 4.1 over both standard method (3.4) and its analogue computed by (4.2).

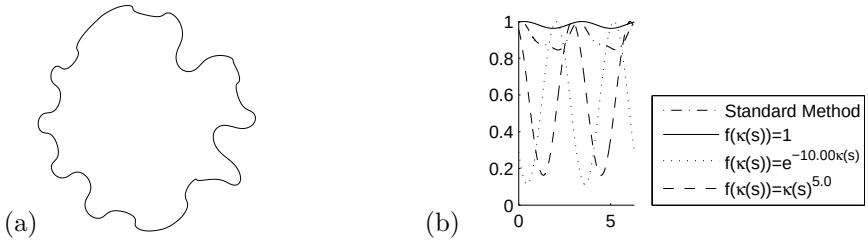
First advantage relates to the situations where some of methods considered fail. As already mentioned, due to the diversity of shapes, it is natural to expect that there are always situations where the method used fails. The standard method for the computation of shape orientation fails if the conditions in (3.5) are satisfied. The method which is an analogue of the standard method, also fails if the corresponding optimizing integral $I(\alpha, \partial S, \rho)$ (see (4.1)) is a constant function, i.e., when

$$\overline{\eta}_{2,0}(\partial S) \; - \; \overline{\eta}_{0,2}(\partial S) = 0 \quad \text{and} \quad \overline{\eta}_{1,1}(\partial S) = 0.$$

A simple idea to overcome such problems was to consider a higher exponent $2N$ in the optimizing integral (for more details see [45, 49]). The problem is that such modified optimizing functions (integrals) do not allow a closed form solution. So, higher exponents involved might be computationally very expensive.

On the other side, Definition 4.1 allows an additional option to overcome situations when the method does not work. Indeed, if the method fails for a certain choice of weighting function, it could work for another choice of them. I.e., if the weighting function $f(\kappa(s))$ is replaced with another weighting function $g(\kappa(s))$, the situation could be changed (for shapes which are not rotationally symmetric), meaning that the new optimizing function $I_g(\alpha, \partial S)$ becomes a nonconstant function, with a well-defined maximum. Then, the optimizing function $I_g(\alpha, \partial S)$ can be used to define a reliable orientation of S.

But a constant optimizing function is not the only problem. The problem is also if the optimizing function has no distinct extreme values. This would imply that the computed orientation is strongly dependent on noise or on the digitization process applied. In such situations a change of the weighting function $f(\kappa(s))$ could lead to more stable and more reliable orientations being computed. This is illustrated by the example in Figure 9. The shape in in Figure 9(a) has no intuitively clear orientation.



standard method	$f(\kappa(s)) = 1$	$f(\kappa(s)) = e^{-10.0\kappa(s)}$	$f(\kappa(s)) = \kappa(s)^{5.0}$
123.0°	17.3°	116.5°	166.8°

FIGURE 9. Graphs of the optimization functions used to orient the shape in (a) are scaled such that their maximum becomes equal to 1, as displayed in (b).

The graphs displayed in Figure 9(b) correspond to the different waiting functions used to compute the orientations. Naturally, if the minima and maxima of the optimizing functions are more distinct then the shape is 'more orientable' (the shape orientability problem is discussed in [53]). I.e., the computed orientation is more reliable. In view of the presented example it can be said that the choice of $f(\kappa(s)) = \kappa(s)^5$ or $f(\kappa(s)) = \kappa(s)^{-10}$ leads to the computed orientations which are more reliable than the orientations computed by the standard method or if the weighting functions $f(\kappa(s)) = 1$ is used. Even that the orientation computed by using $f(\kappa(s)) = \kappa(s)^5$ and $f(\kappa(s)) = \kappa(s)^{-10}$ (166.8° and 116.5° respectively) differ essentially, they are both understood as very reliable because they correspond to the distinct maxima of the corresponding optimizing functions.

As mentioned, the shape in Figure 9(a) has not an intuitively clear orientation but notice that, for certain image processing tasks, this is not a problem. What is crucial is that the computed orientation is reliable. E.g., in a robot manipulation task, all copies of a given product would be positioned consistently if the computed orientation is reliable, independently does the computed orientation match human perception or not.

Another benefit from having a tunable method (i.e., weighting functions, in this case) is illustrated by a turkey shape in Figure 10. For this shape the standard method gives an orientation of 83.6°, but this orientation is not reliable. The optimization function varies between 0.0059 and 0.0065 and the corresponding graph

is rather flat (see the graphs in Figure 10). Since the standard method does not give a distinct orientation, small deviations on the shape boundary (caused by the noise or by the digitization process applied) could lead to an essential deviation of the computed orientation. Indeed, when some noise is added to the shape in Figure 10(a) the computed orientation (by the standard method) changes to 68.8° (Figure 10(b)) and to 97.4° (Figure 10(c)), depending on the noise level. Thus, the standard method is not suitable to be applied to the shape in Figure 10(a) and its "noise" appearances in Figure 10(b) and Figure 10(c). That is in accordance with the values in the table below.

	$\lvert\mu_{1,1}(\partial S)\rvert$	$\lvert\mu_{2,0}(\partial S)-\mu_{0,2}(\partial S)\rvert$
Figure 10(a)	0.000063	0.000658
Figure 10(b)	0.000311	0.000727
Figure 10(c)	0.000086	0.000664

The values of $|\mu_{2,0}(\partial S)-\mu_{0,2}(\partial S)|$ and $|\mu_{1,1}(\partial S)|$ are almost zero, and, consequently, the optimizing function $F(\alpha, S)$ is almost constant. So, it is not possible to distinguish the extreme values of $F(\alpha, S)$ accurately.

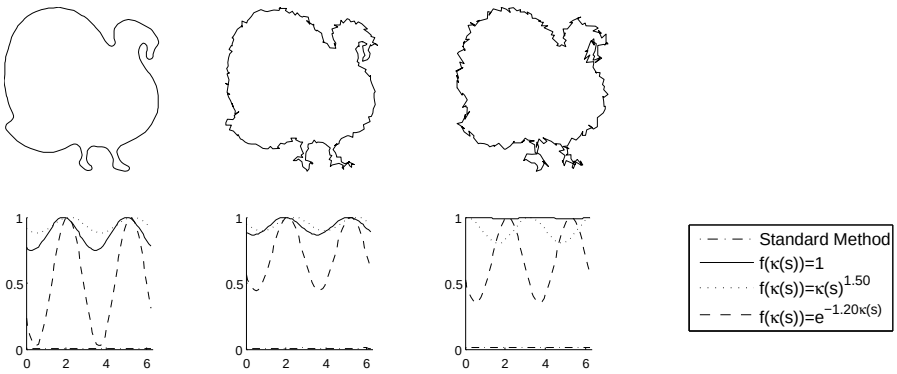
However, a reliable orientation of the shape in Figure 10(a) can be computed by the method given by Definition 4.1. Indeed, if the weighting function is $f(\kappa(s)) = e^{-1.5\kappa(s)}$ then the shapes in Figure 10(a)-(c) are oriented consistently. The reached consistency among the orientations computed is expected because the weighting function $f(\kappa(s)) = e^{-1.5\kappa(s)}$ gives a small weights to high curvature points. In this way the noise effects are minimized.

On the other side, if the weighting functions $f(\kappa(s)) = 1$ and $f(\kappa(s)) = \kappa(s)^{1.5}$ are used, then the computed orientations are highly dependent on the noise–see the table in Figure 10. This is because these weighting functions give reasonably high weights to a high curvature points.

4.3. Convexity Measure. One of the mostly employed shape descriptors is the shape convexity. Over the years many convexity measures have been developed (e.g., [3, 20, 26, 31, 36, 37, 51, 52]) and have been applied to tasks such as image segmentation, object classification, objects recognition, etc. Here we present a boundary based convexity measure developed in [52]. The measure can be applied to both closed and open curves. We start with the following definition of convex curves.

Definition 4.2. A curve γ is convex if and only if for each two points A and B on the curve γ the open line segment (AB) does not intersect the curve γ (i.e., $(AB) \cap \gamma = \emptyset$) or (AB) completely belongs to the curve γ (i.e., $(AB) \subset \gamma$).

It is easy to see that the Definition 4.2 is equivalent to a very common definition of a convex curve which says that a given curve γ is convex if and only if for each point $A \in \gamma$ there is a line l such that $A \in l$ and the curve γ completely lies in one of the closed half planes determined by the line l. Based on Definition 4.2 we are



(a) (b) (c)

	Standard method	$f(\kappa(s)) = 1$	$f(\kappa(s)) = \kappa(s)^{1.5}$	$f(\kappa(s)) = e^{-1.5\kappa(s)}$
(a)	83.6°	106.5°	123.7°	117.9
(b)	68.8°	112.2°	135.1°	117.9
(c)	97.4°	26.3°	3.4°	120.3

FIGURE 10. Orientation of the shape in (a) with the different level of noise added in (b) and (c). The graphs corresponding to the new method (for different weighting functions) are scaled such that their maximum is 1, while the graph of the optimizing function used in the standard method is given on its natural scale. The table includes the computed orientations.

able to define a new convexity measure for single curves, but also for disconnected curves consisting of several arcs.

Definition 4.3. Let $\gamma = \gamma_1 \cup \cdots \cup \gamma_n$ be a curve that consists of $n \geqslant 1$ curve segments, and let A and B be two randomly selected points from γ. The convexity measure $\mathbf{M}(\gamma)$ is defined as the probability that one of the following two events occur:

- the open straight line segment (AB) does not intersect γ (i.e., $(AB) \cap \gamma = \emptyset$), or
- the open straight line segment (AB) completely belongs to γ (i.e., $(AB) \subset \gamma$).

The measure $\mathbf{M}(\gamma)$ has the following desirable properties ($\gamma = \gamma_1 \cup \cdots \cup \gamma_n$ is not a necessarily connected curve):

(a) $\mathbf{M}(\gamma) \in (0, 1]$.
(b) $\mathbf{M}(\gamma) = 1$ if and only if there is a convex curve ρ such that $\gamma \subset \rho$.
(c) $\mathbf{M}(\gamma)$ is invariant under similarity transformations.
(d) for any $\varepsilon > 0$ there is a curve γ such that $\mathbf{M}(\gamma) < \varepsilon$.

A draw-back of $\mathbf{M}(\gamma)$ is that a closed formula for the computation of $\mathbf{M}(\gamma)$ can be computed reasonably easily only in particular cases. More over in most image processing tasks the equation of γ or the equations of γ-segments remain unknown. In such cases it is only possible to estimate $\mathbf{M}(\gamma)$. There are straightforward methods to do this quickly and accurately.

Now, we give several examples to illustrate how $\mathbf{M}(\gamma)$ acts. The first example in Figure 11 demonstrates how $\mathbf{M}(\gamma)$ acts in situations if there is some overlap between objects or there is no a clear difference between foreground and background pixels. In such cases only fragments of the boundary can be extracted; nevertheless, it would still be worth computing shape information from the available data and $\mathbf{M}(\gamma)$ can provide some.

(a) (b) $\mathbf{M} = 0.309$ (c) $\mathbf{M} = 0.767$ (d) $\mathbf{M} = 0.708$

FIGURE 11. (a) Extracting the boundary of `Lena`'s hat is difficult due to poor contrast in places as well as clutter. (b) After edge detection and linking, the edge segments relating to the hat have been manually selected. (c) The outer boundary curves. (c) Gaps between the outer boundary curves completed by straight line segments. The new convexity measure $\mathbf{M}(S)$ can cope with all those situations.

The second example is in Figure 12. Handwritten digits are shown. The ranking the digits according to the convexity measure $\mathbf{M}(\gamma)$ demonstrates a good application potential. Digits "1", "4", "5" and "8" are separated correctly, even that there are substantial natural variations, not only in their overall shape, but also in topology. For instance, one of each of the "0" and "2" digits one example is self intersecting while the other is not. Even small, the example presented indicates that the convexity measure would be a useful property for classification of the digits.

The third example illustrates how $\mathbf{M}(\gamma)$ can be used for the signature recognition tasks. The signatures presented are treated as multiple curve segments–see (see Figure 13). Again noticeable variability is evident within each individual. Considerably lower convexity values are obtained because these curves (i.e., signatures) are more complex than the individual digits in Figure 12. It can be seen from the ranking by $\mathbf{M}(\gamma)$ that convexity is a sufficient descriptor for classification in this small example.

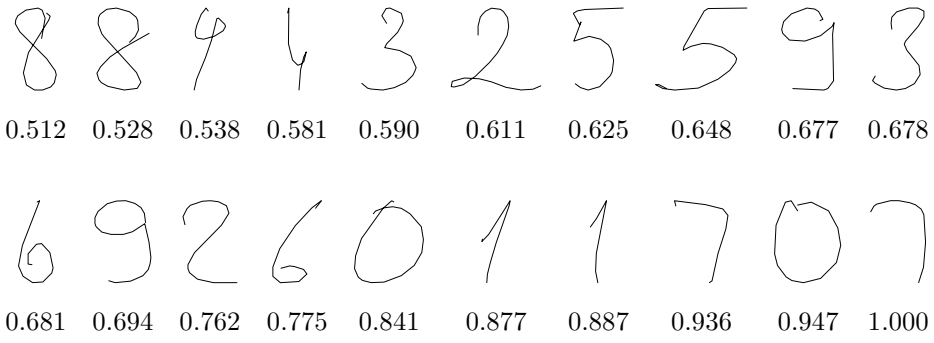


FIGURE 12. Handwritten digits ordered by their $\mathbf{M}(\gamma)$ convexity values.

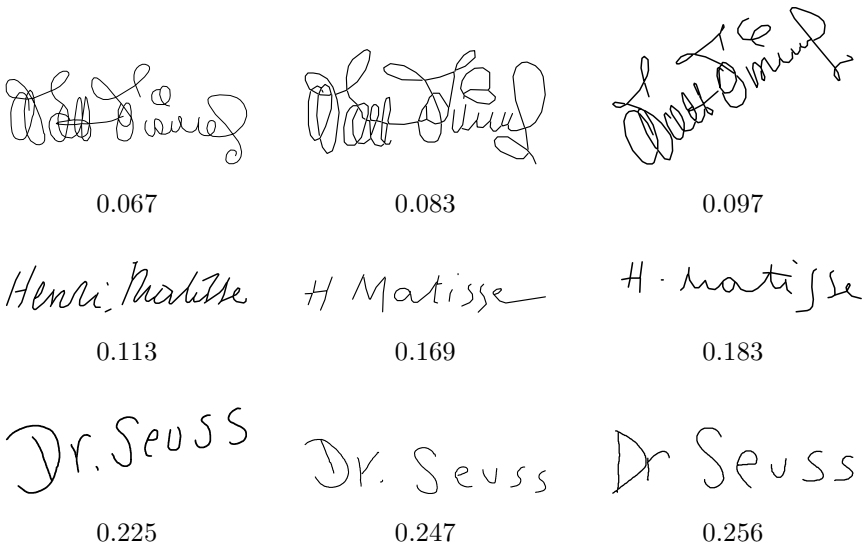


FIGURE 13. Signatures of Walt Disney, Henri Matisse and Dr. Seuss ordered by their $\mathbf{M}(\gamma)$ convexity values.

5. Multi-component Shape Approach

As mentioned, there is no method for the computation of shape orientation which is dominant in all situations. That is a reason why many other methods, different from standard one, are developed. New applications cause new demands for particular method performances. The recent paper [55] has introduced a new method for the computation of shape orientation with a particular request that

method should be applicable to multi-component shapes. The method is described as follows:

- Let S be a given shape, and consider all the line segments $[AB]$ where $A, B \in S$.
- Let $\overrightarrow{a} = (\cos\alpha, \sin\alpha)$ denote the unit vector in the direction α.
- Also, let $\mathbf{pr}_{\overrightarrow{a}}[AB]$ be the projection of the line segment $[AB]$ onto a line parallel to $\overrightarrow{a}$, while $|\mathbf{pr}_{\overrightarrow{a}}[AB]|$ denotes the length of such a projection (for the notations see Figure 14).

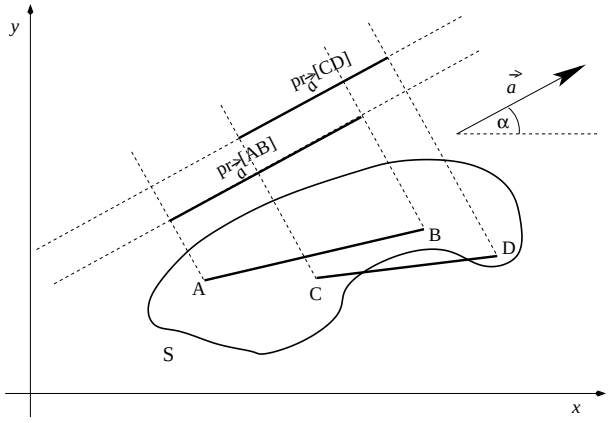


FIGURE 14. Projections of all the line segments whose endpoints lie in S are considered, irrespective of whether the line segment intersects the boundary of S (e.g., the line segment $[CD]$) or not (e.g., $[AB]$).

Then, we define the orientation of the shape S by the direction α that maximizes the integral of the squared lengths of the projections $\mathbf{pr}_{\overrightarrow{a}}[AB]$ onto a line having this direction. A formal definition follows.

Definition 5.1. The orientation of a given shape S is defined by the angle α where the function

$$G(\alpha, S) = \iiiint\limits_{\substack{A=(x,y)\in S \\ B=(u,v)\in S}} |\mathbf{pr}_{\overrightarrow{a}}[AB]|^2 \, dx\, dy\, du\, dv \tag{5.1}$$

reaches its maximum.

Informally speaking, Definition 5.1 defines the orientation of a given shape S by the slope of a line that maximizes the total sum of squared lengths of projections of all straight line segments whose end points belong to S (see Figure 14).

Interestingly, even though Definition 5.1 and the Definition 3.1 come from different motivations, and even the optimizing functions $F(\alpha, S)$ and $G(\alpha, S)$ are different, it has been shown that the orientations computed by these two methods

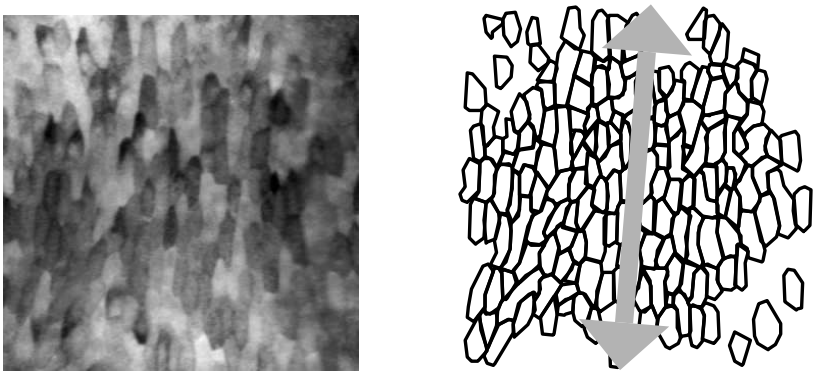


FIGURE 15. An embryonic tissue image (on the left) is presented as a multi-component shape (on the right). The desired computed orientation is presented by the arrow.

are the same. Indeed, Theorem 5.1 (for a proof see [55]) shows that for a fixed shape S the quantity $G(\alpha, S) + 2 \cdot m_{0,0}(S) \cdot F(\alpha, S)$ is the same for all $\alpha \in [0, 2\pi)$. Furthermore, $G(\alpha, S) + 2 \cdot m_{0,0}(S) \cdot F(\alpha, S) = \text{constant}$ implies that the maximum of $G(\alpha, S)$ and minimum of $F(\alpha, S)$ are reached at the same points, i.e., the angle where $G(\alpha, S)$ reaches the maximum must be the angle where $F(\alpha, S)$ reaches its minimum. So, the shape orientations computed by Definition 3.1 and Definition 5.1 coincide.

Theorem 5.1. *Let a shape S be given. Then*

$$G(\alpha, S) + 2 \cdot m_{0,0}(S) \cdot F(\alpha, S) = 2 \cdot m_{0,0}(S) \cdot (\overline{m}_{2,0}(S) + \overline{m}_{0,2}(S))$$

is true for all $\alpha \in [0, 2\pi]$.

Definition 5.1 has an essential advantage over Definition 3.1 because it has a natural extension to compound (i.e., multi-component) shapes. Just to mention that the dealing with multi-component objects is of a great importance because in many situations, several single objects usually appear as a group and act together (e.g., blood cells, vehicles on the road, fish shoal or group of people (as in Figure 16), etc). Also, in many situations, it is suitable to consider a single object as a multi-component one, consisting of many components defined with respect to some natural criteria (as an embryonic tissue displayed in Figure 15, or materials micro structures, wood textures, etc). Further, a sequence of the same object appearing on a sequence of frames (e.g. a walking human in Figure 18) can be understood as as a multicomponent shape.

Surprisingly, the orientation of multi-component shapes has not been intensively studied in literature yet. Recently, [55] has introduced a method for computation of the orientation of multi-component shapes, presented in 2D images. The method is theoretically well founded, and thus, its behavior can be well understood, but also predicted to some extent, which is always an advantage.

It is worth mentioning that there is no an easy and straightforward way to compute the orientation of a multi-component shape from the orientations (computed by some of the existing methods for single component shapes) of its components. Indeed, a very natural idea would be to compute the orientation of a multi-component shape from the orientations assigned to its components, probably weighted by some coefficients computed from the 'component importance' (e.g. the component size or its relative position inside the shape as whole). However, the problem is that a huge majority of methods define the shape orientation by a line (not by a vector). This implies an ambiguity of 180 degrees (e.g. the computed orientations α and $\alpha + 180^\circ$ are assumed to be the same). So, if $S_1, S_2, \ldots, S_n$ are components of a multi-component shape S, then most of the existing methods would compute their orientations as $\varphi_1 + a_1 \cdot 180^\circ$, $\varphi_2 + a_2 \cdot 180^\circ, \ldots, \varphi_n + a_n \cdot 180^\circ$, where the numbers $a_1, a_2, \ldots, a_n$ are arbitrarily chosen from $\{0, 1\}$. Thus if, in the simplest variant, the orientation of multi-component shape $S = S_1 \cup S_2 \cup \cdots \cup S_n$ is computed as the average value of the orientations assigned to its components, then the orientation of S would be computed as

$$\frac{(\varphi_1 + a_1 \cdot 180^\circ) + \ldots + (\varphi_n + a_n \cdot 180^\circ)}{n} = \frac{\varphi_1 + \ldots + \varphi_n}{n} + \frac{(a_1 + \ldots + a_n) \cdot 180^\circ}{n}$$

and obviously, for different choices of $a_1, a_2, \ldots, a_n$, the computed orientations are inconsistent (i.e., they could differ for an arbitrary multiple of the fraction $180^\circ/n$). This is obviously unacceptable.

As mentioned, Definition 5.1 allows a straightforward extension to the multi-component shapes, as given by the following definition.

Definition 5.2. Let S be a compound shape which consists of m disjoint shapes $S_1, S_2, \ldots, S_m$. Then the orientation of S is defined by the angle that maximizes the function $G_{comp}(\alpha, S)$ defined by

$$G_{comp}(\alpha, S) = \sum_{i=1}^{m} \iiiint\limits_{\substack{A=(x,y)\in S_i \\ B=(u,v)\in S_i}} |\mathbf{pr}_{\vec{a}}[AB]|^2 dx\, dy\, du\, dv. \tag{5.2}$$

The previous definition enables an easy computation of the orientation of compound objects, as shown by the following theorem (see [55] for the proof details).

Theorem 5.2. *Let a compound shape S, consisting of m disjoint shapes $S_1, S_2, \ldots, S_m$, be given, and let the function $G_{comp}(\alpha, S)$ be defined as in* (5.2).

The angle α where the function $G_{comp}(\alpha, S)$ reaches its maximum satisfies the following equation

$$\frac{\sin(2\alpha)}{\cos(2\alpha)} = \frac{2 \cdot \sum_{i=1}^{m} \overline{m}_{1,1}(S_i) \cdot m_{0,0}(S_i)}{\sum_{i=1}^{m} (\overline{m}_{2,0}(S_i) - \overline{m}_{0,2}(S_i)) \cdot m_{0,0}(S_i)}. \tag{5.3}$$

The new method has some specific properties which appear to be very desirable when computing the orientation of multi-component shapes. These properties do not hold if the 'single component' methods are applied to multi-component shapes. Some of such properties are listed below as separate remarks.

Remark 5.1. Any component S_i, of a compound shape $S = S_1 \cup \cdots \cup S_m$, which cannot be oriented by optimizing $G(\alpha, S_i)$ (i.e., $G(\alpha, S_i) = \text{constant}$) will not contribute to (5.3), and is therefore ignored in the computation of $G_{comp}(\alpha, S)$. That is because $G(\alpha, S_i) = \text{constant}$ implies $\overline{m}_{1,1}(S_i) = 0$ and $\overline{m}_{2,0}(S_i) = \overline{m}_{0,2}(S_i)$.

Remark 5.2. If all components of S have identical orientation according to $G(\alpha, S)$ then this same orientation is also computed by $G_{comp}(\alpha, S)$.

The weighting given to the shape components in (5.3) causes a big influence of the larger components to the computed orientation. For instance, let a compound shape S which consists of shapes S_1 and S_2' such that the shape S_2' is the dilation of a shape S_2 by a factor $\mathbf{r}$, i.e., $S_2' = \mathbf{r} \cdot S_2 = \{(\mathbf{r} \cdot x, \mathbf{r} \cdot y) \mid (x, y) \in S_2\}$. Then,

$$m_{0,0}(S_2') = \mathbf{r}^2 \cdot m_{0,0}(S_2),\ \overline{m}_{1,1}(S_2') = \mathbf{r}^4 \cdot \overline{m}_{1,1}(S_2),$$
$$\overline{m}_{2,0}(S_2') = \mathbf{r}^4 \cdot \overline{m}_{2,0}(S_2),\ \overline{m}_{0,2}(S_2') = \mathbf{r}^4 \cdot \overline{m}_{0,2}(S_2).$$

Entering these estimates into (5.3) we obtain that the orientation α of the compound shape $S = S_1 \cup S_2'$ should be computed from

$$\begin{aligned}(5.4)\quad &\frac{\sin(2\alpha)}{\cos(2\alpha)} \\ &= \frac{2 \cdot \overline{m}_{1,1}(S_1) \cdot m_{0,0}(S_1) + 2 \cdot \overline{m}_{1,1}(S_2') \cdot m_{0,0}(S_2')}{(\overline{m}_{2,0}(S_1) - \overline{m}_{0,2}(S_1)) \cdot m_{0,0}(S_1) + (\overline{m}_{2,0}(S_2') - \overline{m}_{0,2}(S_2')) \cdot m_{0,0}(S_2')} \\ &= \frac{2 \cdot \overline{m}_{1,1}(S_1) \cdot m_{0,0}(S_1) + 2 \cdot \mathbf{r}^6 \cdot \overline{m}_{1,1}(S_2) \cdot m_{0,0}(S_2)}{(\overline{m}_{2,0}(S_1) - \overline{m}_{0,2}(S_1)) \cdot m_{0,0}(S_1) + \mathbf{r}^6 \cdot (\overline{m}_{2,0}(S_2) - \overline{m}_{0,2}(S_2)) \cdot m_{0,0}(S_2)}.\end{aligned}$$

Obviously, the influence of S_2' to the computed orientation of S could be very big, if the dilation factor $\mathbf{r}$ is much bigger than 1. This suggests a modification of (5.3) to enforce different weighting (as a function of the components area), assigned to the shape components

$$(5.5)\qquad \frac{\sin(2\alpha)}{\cos(2\alpha)} = \frac{2 \cdot \sum_{i=1}^{m} \overline{m}_{1,1}(S_i)/m_{0,0}(S_i)}{\sum_{i=1}^{m} (\overline{m}_{2,0}(S_i) - \overline{m}_{0,2}(S_i))/m_{0,0}(S_i)}.$$

If the orientation α of $S = S_1 \cup S_2' = S_1 \cup \mathbf{r} \cdot S_2$ is computed by (5.5) then it satisfies

$$\begin{aligned}&\frac{\sin(2\alpha)}{\cos(2\alpha)} \\ &\quad = \frac{2 \cdot \overline{m}_{1,1}(S_1)/m_{0,0}(S_1) + 2 \cdot \mathbf{r}^2 \cdot \overline{m}_{1,1}(S_2)/m_{0,0}(S_2)}{(\overline{m}_{2,0}(S_1) - \overline{m}_{0,2}(S_1))/m_{0,0}(S_1) + \mathbf{r}^2 \cdot (\overline{m}_{2,0}(S_2) - \overline{m}_{0,2}(S_2))/m_{0,0}(S_2)}.\end{aligned}$$

So, the impact of an increase of $\mathbf{r}$ to both nominator and denominator is smaller than if (5.3) is applied directly (see (5.4)).

Further, it is not difficult to imagine situations where the size of components should have no effect on the computed shape orientation. For instance, objects (i.e., components of a compound object) may be of the same size in nature, but appear as objects of a different size in the image due to varying distances from the

camera. If we would like to avoid any impact of the size of the components to the computed orientation, then we can use the following formula

$$\frac{\sin(2\alpha)}{\cos(2\alpha)} = \frac{2\cdot\sum_{i=1}^{m}\overline{m}_{1,1}(S_i)/(m_{0,0}(S_i))^2}{\sum_{i=1}^{m}(\overline{m}_{2,0}(S_i)-\overline{m}_{0,2}(S_i))/(m_{0,0}(S_i))^2}. \tag{5.6}$$

In the view of the previous simple example, the computed orientation of $S = S_1 \cup S_2' = S_1 \cup \mathbf{r}\cdot S_2$ is the same for each $\mathbf{r} > 0$. Indeed, if the last formula is applied then the computed orientation α satisfies

$$\frac{\sin(2\alpha)}{\cos(2\alpha)} = \frac{2\cdot\overline{m}_{1,1}(S_1)/(m_{0,0}(S_1))^2 + 2\cdot\overline{m}_{1,1}(S_2)/(m_{0,0}(S_2))^2}{(\overline{m}_{2,0}(S_1)-\overline{m}_{0,2}(S_1))/(m_{0,0}(S_1))^2 + (\overline{m}_{2,0}(S_2)-\overline{m}_{0,2}(S_2))/(m_{0,0}(S_2))^2}.$$

Thus, $\mathbf{r}$ does not have any impact to the computed orientation.

The behaviour of the shape orientation method, based on the new multi-component approach, introduced in [55] is demonstrated on several examples.

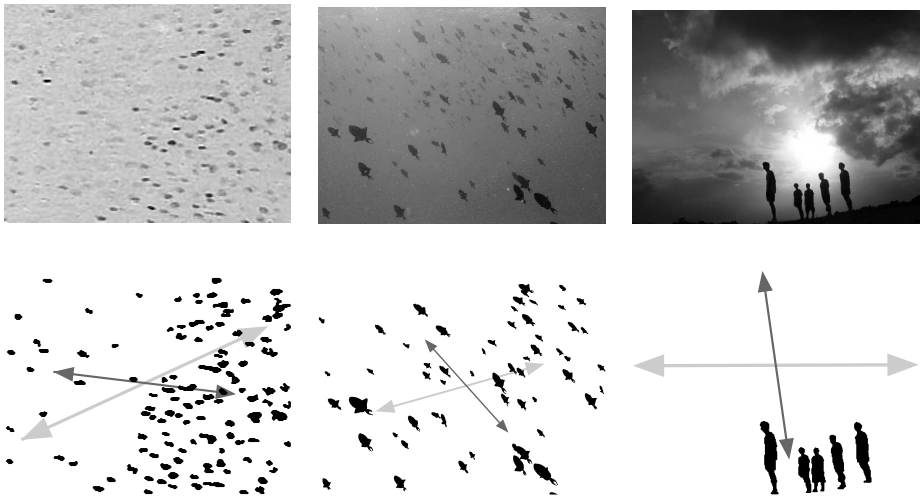
First, we consider examples in Figure 16. Obvious difficulties are apparent in Figure 16(a) in which most of the components are at best only moderately oriented, while many have no distinct orientation, leading to considerable variability in individual orientation estimates. Nevertheless, the overall orientation (formula (5.5) is used) is correctly determined.

In Figure 16(b) a shoal is presented. The orientation of most of fish in the shoal is well defined. Fish orientations (in the most cases) are coincident with the direction of shoal motion. The same orientation is obtained if the formula (5.5) is applied to the shoal as a compound object. Notice that the standard method gives an unacceptable shoal orientation.

Similar discussion holds for the silhouettes of men presented in Figure 16(c).

Images in Figure 17 demonstrates an interesting and very desirable property of the multi-component approach. The central pair of arrows are the orientations shown previously in Figure 16. In addition, the each image was split into two parts by a vertical line, and the orientations were calculated separately for each sub-set of components. Whereas the the new method produces a consistent orientation in both cases, the standard method's orientation varies considerably and does not much our perception.

The third example is in Figure 18 and is related to the, so called, outlier detection problem. More precisely, the application is gait recognition and the binary data is taken from the NLPR Gait Database [46]. The binary masks were generated and provided by Wang et al. [46] using background subtraction but, as noted by the authors, many segmentation errors remain. This causes that the silhouettes are often fragmented into multiple components. Although most of these can be readily corrected using standard morphological operations there remain larger errors that would need to be identified and processed separately. To apply the multi-component shape approach here, the set of blobs in each image frame is considered as a single component. So far, we considered multiple components distributed spatially within a single image, in this application the multiple components are distributed



(a) blood cells (b) shoal (c) people

FIGURE 16. Presented objects are treated as components of a multi-component shape. Orientations were computed by (5.5)–shorter black arrows, and by the standard method–long gray arrows.

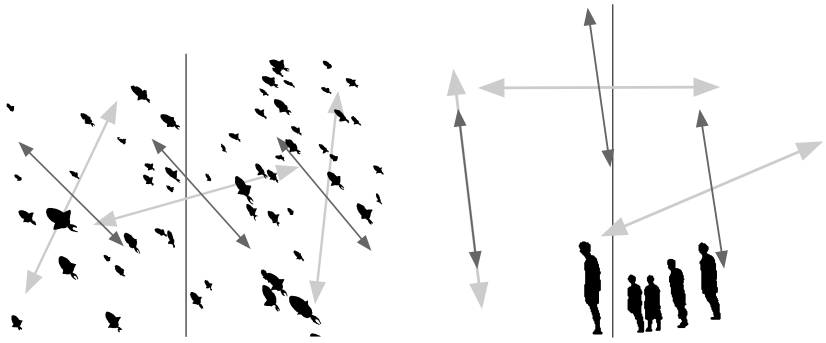


FIGURE 17. Orientations computed separately for the left and right halves of the images, and also for the complete images. Orientations computed by (5.5) are given by shorter dark arrows; orientations computed by the standard method are given by long light arrows.

temporally across the sequence of images. Since we assume that all frames in the sequence have the same importance, the weighting for each component is set to be independent of size, and therefore (5.6) is used to compute orientation. Also, a natural expectation is that if the components are fairly consistently oriented, then

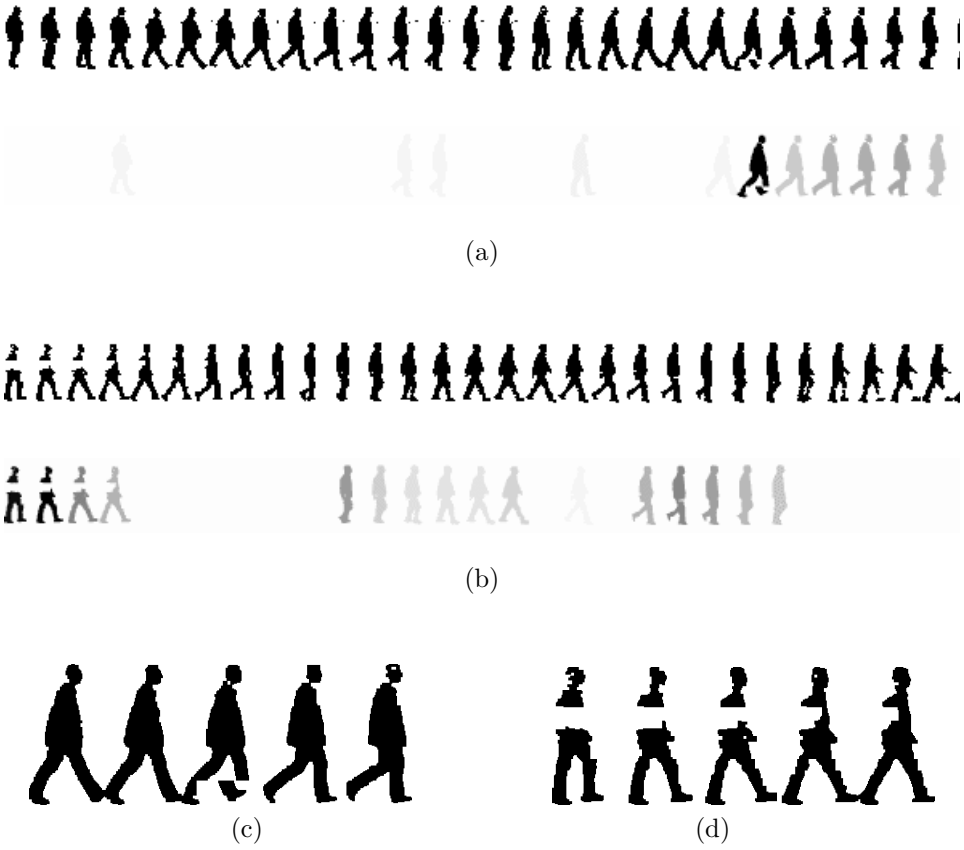
faulty segmentation is likely to result in atypical component orientations. Two examples are given in Figure 18. The difference in orientation caused by removing the least consistent component (i.e., image frame) is computed for up to half the number of components, and the frames are replotted with darkness proportional to their difference values. The remaining half of the frames are considered as inliers and their differences are ignored. It can be seen that in Figure 18(a) there has been some kind of error in the original processing chain that produced the binary images, and the person's leading leg has been displaced. In Figure 18(b) the quality of change detection at the beginning of the sequence is poor, which is more visible from Figure 18(c) and Figure 18(d). In both cases these segmentation errors have been identified as orientation outliers (indicated by the darker frames below). In Figure 18(b) there is a second instance of poor change detection two thirds of the way through the sequence, which has not been clearly identified as containing outliers.

6. Conclusion

In this article we have given an overview of some of standard shape based technique used in object classification, object recognition and object identification tasks. Also, some recent developments were discussed, mainly those introduced by the author and his collaborators. We have started with area based shape descriptors, which were most popular in the past because of their robustness and simplicity. Moments, Hu moment invariants, shape elongation, and the standard method for the computation of shape orientation are overviewed. A new circularity measure, derived from the first Hu moment invariant is also studied and its extension to a family of circularity measures is introduced. This illustrates that the classification efficiency can be improved by using several measures devoted to estimate the same shape property (in this case, the shape circularity). Illustrative experiments are provided in order to explain how the methods presented work.

The next, attention has been paid to the boundary based shape descriptors. Boundary based shape descriptors become more popular, in the recent days, because they are more sensitive and can be used in high precision tasks. Particularly, they are suitable for object and person identification tasks, which is very important due to a strong demand for high precision identification tools (e.g., in crime prevention applications). Some area based shape descriptors can be easily extended to their boundary based analogues (e.g., Hu moment invariants [9]), but this is not always possible. It is worth mentioning that boundary based shape descriptors have additional advantages over area based ones. For example, boundary based descriptors are able to cope with objects with particularly extracted boundaries, they are usually faster to compute, and they can be applied to the objects which are linear in their nature (digits, signatures, face details, etc).

Very often, different applications require different method performances. Thus, tuning possibilities are always welcomed when a shape descriptor/measure is created. It has been described here how to define the shape orientation by using the curvature (at the shape boundary points) as a possible tuning parameter. Several



(a)

(b)

(c) (d)

FIGURE 18. Two gait sequences. (a),(b) For each sequence the extracted silhouettes are displayed and underneath is an intensity coding of each silhouette to show its degree of being an outlier (dark means high likelihood). (c),(d) a magnified view of the most outlying silhouette from each sequence and its neighbours.

benefits are obtained. E.g., the behavior of a particular measure can be controlled, more distinct shape orientations can be computed, shapes which are not orientable by certain methods become orientable when use a suitable choice of the tuning parameters, etc. In addition, we have discussed one of convexity measures which can be applied to open curve segments. Simple examples which illustrate the applicability of such a convexity measure to the digits recognition and signature classification tasks are given.

Finally, a recent multi-component shape analysis approach is discussed. Namely, very often it is better to consider a group of objects as a single multi-component object (fish shoal, vehicles on a road, etc). Also, sometimes is more convenient to treat a single object, as a multi-component one, consisting of naturally defined

components (e.g., cellular materials like bones, artificially tailored materials, words or signatures decomposed onto characters, etc). Another possibility is to consider an appearance of the same object in a frame sequence as a multi-component object. An application to the detecting outliers, in a sequence of images, is given. In a similar way, an unusual behavior of a person can be detected (could be of an interest in the crime prevention).

To close this overview article, the author believes that challenges and possibilities for further performance improvements, in object classification, recognition and identifications applications, lie in new boundary based approaches, rather than in the area based ones.

Due to the luck of space, $3D$ shape descriptors are not discussed. Just to mention that the recent developments in image technologies made $3D$ data widely available. Also, methods for the reconstruction of $3D$ objects from the corresponding $2D$ images are already well established. Notice that shape descriptor techniques are particularly suitable when working in $3D$ space. This is because shape descriptors, as global $3D$ object features [5, 25, 28], require much less time for the processing than local based features do (simply, there can be too many of them when working in $3D$ space).

Acknowledgement. This work is partially supported by the Serbian Ministry of Education and Science, the projects ON174008, "Advanced Techniques of Cryptology, Image Processing and Computational Topology for Information Security", and III44006, "Development of new information and communication technologies, based on advanced mathematical methods, with applications in medicine, telecommunications, power systems, protection of national heritage and education".

References

[1] N. Alajlan, M. S. Kamel, G. H. Freeman, *Geometry-based image retrieval in binary image databases*, IEEE Trans. Patt. Anal. Mach. Intell. **30** (2008), 1003–1013.

[2] E. T. Bowman, K. Soga, T. Drummond, *Particle shape characterization using Fourier analysis*, Geotechnique **51** (2001), 545–554.

[3] L. Boxer, *Computing deviations from convexity in polygons*, Pattern Recognition Letters **14** (1993), 163–167.

[4] J. M. H. du Buf, M. M. Bayer (eds), *Automatic Diatom Identification*, World Scientific, 2002.

[5] E. Bustos, D. A. Keim, D. Saupe, T. Schreck, D. V. Vranic, *Feature-based similarity search in object databases*, ACM Comput. Surv. **37** (2005), 345–387.

[6] J. Cortadellas, J. Amat, F. De la Torre, *Robust normalization of silhouettes for recognition applications*, Pattern Recognition Letters **25** (2004), 591–601.

[7] D. Coeurjolly, R. Klette, *A comparative evaluation of length estimators of digital curves*, IEEE Trans. Patt. Anal. Mach. Intell. **26** (2004), 252–257.

[8] T. F. Cootes, C. J. Taylor, D. H. Cooper, J. Graham, *Active shape models-their training and application*, Computer Vision and Image Understanding **61** (1995), 38–59.

[9] C.-C. Chen, *Improved moment invariants for shape discrimination*, Pattern Recognition **6** (1993), 683-686.

[10] H. Freeman, R. Shapira, *Determining the minimum-area encasing rectangle for an arbitrary closed curve*, Comm. ACM **18** (1975), 409–413.

[11] G. H. Granlund, *Fourier preprocessing for hand print character recognition*, IEEE Trans. Computers **21** (1972), 195–201.

[12] J. Flusser, T. Suk, *Rotation moment invariants for recognition of symmetric objects*, IEEE Transactions Image Processing, textbf15 (2006), 3784–3790.
[13] J. Flusser, J. Kautsky, F. Sroubek, *Implicit moment invariants*, International J. of Computer Vision **86** (2010), 72–86.
[14] V. H. S. Ha, J. M. F. Moura, *Affine-permutation invariance of 2-D shapes*, IEEE Transactions Image Processing **14** (2005), 1687–1700.
[15] R. M. Haralick, *A measure for circularity of digital figures*, IEEE Transactions Systems, Man and Cybernetics **4** (1974), 394–396.
[16] M. Hu, *Visual pattern recognition by moment invariants*, IRE Trans. Inf. Theory **8** (1962), 179–187.
[17] M. N. Huxley, *Exponential sums and lattice points. III*, Proc. London Math. Soc. **87** (2003), 591–609.
[18] A. K. Jain, A. Vailaya, *Shape-based retrieval: A case-study with trademark image databases*, Pattern Recognition **31** (1998), 1369–1390.
[19] X. Y. Jiang, H. Bunke, *Simple and fast computation of moments*, Pattern Recognition **24** (1991), 801–806.
[20] R. Kakarala, *Testing for convexity with Fourier descriptors*, Electronics Letters **34** (1998), 1392–1393.
[21] R. Klette, A. Rosenfeld,*Digital Geometry*, Morgan Kaufmann, San Francisco, 2004.
[22] R. Klette, J. Žunić, *On Discrete Moments of Unbounded Order*, LNCS **4245** (2006), 367–378.
[23] L. J. Latecki, R. Lakämper, U. Eckhardt. *Shape descriptors for non-rigid shapes with a single closed contour*, In Proc. Conf. Computer Vision Pattern Recognition, pp. 1424–1429, 2000.
[24] J.-G. Leu, *Computing a shape's moments from its frontier*, Pattern Recognition **24** (1991), 949–957.
[25] Z. Lian, P. L. Rosin, X. Sun, *Rectilinearity of 3D meshes*, Int. J. Comput. Vis. **89** (2010), 130-151.
[26] R. R. Martin, P. L. Rosin, *Turning shape decision problems into measures*, Int. J. Shape Modelling **10** (2004), 83-113.
[27] C. Martinez-Ortiz, J. Žunić, *Curvature weighted gradient based shape orientation*, Pattern Recognition **43** (2010), 3035–3041.
[28] C. Martinez-Ortiz, J. Žunić, *Tunable cubeness measures for 3D shapes*, Pattern Recognition Letters **32** (2001), 1871–1881.
[29] D. L. Page, A. Koschan, S. R. Sukumar, B. Roui-Abidi, M. A. Abidi, *Shape analysis algorithm based on information theory*, In Int. Conf. Image Processing, volume 1, pp. 229–232, 2003.
[30] D. Proffitt, *The measurement of circularity and ellipticity on a digital grid*, Pattern Recognition **15** (1982), 383–387.
[31] E. Rahtu, M. Salo, J. Heikkilä, *A new convexity measure based on a probabilistic interpretation of images*, IEEE Trans. Patt. Anal. Mach. Intell. **28** (2006), 1501–1512.
[32] R. M. Rangayyan, N. M. Elfaramawy, J. E. L. Desautels, O. A. Alim, *Measures of acutance and shape for classification of breast-tumors*, IEEE Transactions Medical Imaging **16** (1997), 799-810.
[33] P. L. Rosin, *Measuring shape: Ellipticity, rectangularity, and triangularity*, Machine Vision and Applications **14** (2003), 172–184.
[34] P. L. Rosin, *Computing global shape measures*, In C. H. Chen and P. S. P. Wang, eds, *Handbook of Pattern Recognition and Computer Vision*, pp. 177–196, World Scientific, 2005.
[35] P. L. Rosin, *A two-component rectilinearity measure*, Computer Vision and Image Understanding **109** (2008), 176–185.
[36] P. L. Rosin, C. L. Mumford. *A symmetric convexity measure*, Computer Vision and Image Understanding **103** (2006), 101–111.
[37] P. L. Rosin, J. Žunić, *Probabilistic convexity measure*, IET Image Processing **1** (2007), 182–188.
[38] N. Sladoje, J. Lindblad, *High precision boundary length estimation by utilizing gray-level information*, IEEE Trans. Patt. Anal. Mach. Intell. **31** (2009), 357–363.

[39] D. Shen, H. H. S. Ip, *Optimal axes for defining the orientations of shapes*, Electronic Letters **32** (1996), 1873–1874.
[40] M. Sonka, V. Hlavac, R. Boyle, *Image processing, analysis, and machine vision*, Thomson-Engineering, 2007.
[41] H. I. Stern, *Polygonal entropy: a convexity measure*, Pattern Recognition Letters **10** (1989), 229–235.
[42] M. Stojmenović, J. Žunić, *Measuring linearity of planar point sets*, Pattern Recognition **41** (2008), 2503–2511.
[43] M. Stojmenović, J. Žunić. *Measuring elongation from shape boundary*, J. of Mathematical Imaging and Vision **30** (2008), 73–85.
[44] H. Süsse, F. Ditrich, *Robust determination of rotation-angles for closed regions using moments*, In Int. Conf. Image Processing, volume 1, pp. 337–340, 2005.
[45] W. H. Tsai, S. L. Chou, *Detection of generalized principal axes in rotationally symetric shapes*, Pattern Recognition **24** (1991), 95–104.
[46] L. Wang, T. N. Tan, W. M. Hu, H. Z. Ning, *Automatic gait recognition based on statistical shape analysis*, IEEE Trans. Image Processing **12** (2003), 1120–1131.
[47] D. Xu, H. Li, *Geometric moment invariants*, Pattern Recognition **41** (2008), 240–249.
[48] H. Zabrodsky, S. Peleg, D. Avnir, *Symmetry as a continuous feature*, IEEE Trans. Patt. Anal. Mach. Intell. **17** (1995), 1154–1166.
[49] J. Žunić, L. Kopanja, J. E. Fieldsend, *Notes on shape orientation where the standard method does not work*, Pattern Recognition **39** (2006), 856–865.
[50] J. Žunić, P. L. Rosin, *Rectilinearity measurements for polygons*, IEEE Trans. Patt. Anal. Mach. Intell. **25** (2003), 1193–1200.
[51] J. Žunić, P. L. Rosin, *A new convexity measurement for polygons*, IEEE Trans. Patt. Anal. Mach. Intell. **26** (2004), 923–934.
[52] J. Žunić, P. L. Rosin, *Convexity measure for shapes with partially extracted boundaries*, Electronics Letters **43** (2007), 380–382.
[53] J. Žunić, P. L. Rosin, L. Kopanja, *On the orientability of shapes*, IEEE Trans. Image Processing **15** (2006), 3478–3487.
[54] J. Žunić, M. Stojmenović, *Boundary based shape orientation*, Pattern Recognition **41** (2008), 1785–1798.
[55] J. Žunić, P. L. Rosin, *An alternative approach to computing shape orientation with an application to compound shapes*, Int. J. Comput. Vision **81** (2009), 138–154.
[56] J. Žunić, K. Hirota, P. L. Rosin, *A Hu invariant as a shape circularity measure*, Pattern Recognition **43** (2010), 47–57.