Expert Systems With Applications 89 (2017) 362–373

Contents lists available at ScienceDirect

Expert Systems With Applications

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

Angular descriptors of complex networks: A novel approach for boundary shape analysis

Leonardo F.S. Scabini[a,*], Danilo O. Fistarol[b], Sávio V. Cantero[b], Wesley N. Gonçalves[b], Bruno Brandoli Machado[b,c], Jose F. Rodrigues, Jr.[c]

[a] *Instituto de Física de São Carlos, Universidade de São Paulo, Av. Trabalhador São-Carlense, 400, Parque Arnold Schimidt, 13566-590, São Carlos - SP, Brazil*
[b] *Federal University of Mato Grosso do Sul, Rua Itibiré Vieira, s/n, Residencial Julia Oliveira Cardinal, 79907-414, Ponta Porã - MS, Brazil*
[c] *Instituto de Ciências Matemáticas e Computação, Universidade de São Paulo, Av. Trabalhador São-Carlense, 400, Centro, 13566-590, São Carlos - SP, Brazil*

ARTICLE INFO

Article history:
Received 11 August 2016
Revised 2 August 2017
Accepted 3 August 2017
Available online 3 August 2017

Keywords:
Shape analysis
Complex networks
Computer vision
Feature extraction
Classification

ABSTRACT

We introduce a method for shape recognition based on the angular analysis of Complex Networks. Our method models shapes as Complex Networks defining a more descriptive representation of the inner angularity of the shape's perimeter. The result is a set of measures that better describe shapes if compared to previous approaches that use only the vertices' degree. We extract the angle between the Complex Network edges, and then we analyze their distribution along with a network dynamic evolution. The proposed approach, named Angular Descriptors of Complex Networks (ADCN), presents a high discriminatory power, as evidenced by experiments conducted in five datasets. It is rotation invariant, presents high robustness against scale changes and degradation levels, overcoming traditional methods such as Zernike moments, Multiscale Fractal dimension, Fourier, Curvature and the degree-based descriptors of Complex Networks.

1. Introduction

In a general sense, shape is a preeminent property of almost every entity in the physical world. A large range of objects can be discriminated by their shape, including not only natural circumstances but also synthetic phenomena. In computer vision and pattern recognition, the shape is considered one of the most important features for the identification and distinction of objects in various scenarios (Loncaric, 1998). The principle of shape-oriented analysis is to extract information able to characterize the elements of specific domains, leading to methods used in a wide scope of applications, in areas such as medicine (Shen, Rangayyan, & Desautels, 1994), biology (Neto, Meyer, Jones, & Samal, 2006), and people tracking (Wang, Tan, Hu, & Ning, 2003).

Shape descriptors can be divided into two main categories: those based on contours, and those based on regions (Mehtre, Kankanhalli, & Lee, 1997; Zhang & Lu, 2004). Contour-based approaches focus on extracting information located at the edges of the shape, while region-based approaches consider all the pixels of a certain region of the shape. Among the contour-based techniques, we can cite descriptors based on the Fourier transform (Persoon & Fu, 1977), on Curvature (Wu & Wang, 1993), on Zernike moments (Zhenjiang, 2000), on multiscale fractal dimension (Torres, Falcão, & Costa, 2004), and on multiscale triangle representation (Mouine, Yahiaoui, & Verroust-Blondet, 2013). The region-based descriptors are based on Zernike moments (Kim & Kim, 2000), on invariant moments (Chen & Tsai, 1993), and on histograms of gradients (Xiao, Hu, Zhang, & Wang, 2010), to name a few.

In the last decade, the concept of Complex Network (CN) has been employed in the field of shape analysis, as described in the work of Backes, Casanova, and Bruno (2009). In this approach, a shape is represented by a CN in which each pixel corresponds to a vertex, and the Euclidean distance, along with a threshold, is used to define edges between pairs of vertices. Over one such CN, it becomes possible to use several methods and metrics from Graph Theory in order to characterize the shapes. In the work of Backes et al., the authors introduce a descriptor based on vertex degree; they use the max and the average mean degree as global features. This descriptor, though simple, presented promising results, corroborating that CN has potential in pattern recognition.

In the present work, we proposed a new shape descriptor based on CNs and geometric concepts. Our method can be applied after the image segmentation step; so, given the shape boundary pixels, we build a CN. We explore the fact that the use of global

[*] Corresponding authors.
E-mail addresses: scabini@ifsc.usp.br (L.F.S. Scabini), danilofistarol@gmail.com (D.O. Fistarol), savio.cantero@gmail.com (S.V. Cantero), wesley.goncalves@ufms.br (W.N. Gonçalves), bruno.brandoli@ufms.br (B.B. Machado), junio@usp.br (J.F. Rodrigues, Jr.).

http://dx.doi.org/10.1016/j.eswa.2017.08.009
0957-4174/© 2017 Elsevier Ltd. All rights reserved.

measures limits the discriminating potential of CN-based descriptors. In a different course of action, we contribute by introducing a shape characterization that takes into account the angle between the connected vertices (pixels), a local feature that brings a finer discrimination to the descriptor. We also describe how to automatically set the threshold used to define the edges, which presented superior results compared to the use of constant threshold values. As shown in the experiments, the proposed methodology improved the classification results in most of the cases, including with variations on rotation, scale, noise, and contour degradation. We compared our results to traditional contour and region-based descriptors, and to other CN descriptors.

The paper is organized as follows. Section 2 presents an overview of the CN theory and its application for shape representation. The proposed approach is defined in Section 3, where our techniques for angular analysis and automatic threshold selection are presented. Section 4 details all the experimental protocol, the datasets and the classification results achieved in each experiment, along with the comparison to other descriptors. Finally, in Section 6 an overview of the angular descriptors and its results are presented.

2. Complex network theory

CN has emerged in the past decade combining concepts from graph theory and statistics (Costa, Rodrigues, Travieso, & Villas Boas, 2007). It is a research field that heavily relies on mathematics, computer science, and physics, leading to a large range of applications (Costa et al., 2011). The popularity of CNs can be explained by its ease in modeling many kinds of problems and natural phenomena. As CNs are represented by graphs, every entity-relationship problem can be straightly modeled, such as social interaction, physics simulation, or image representation.

We can cite three main developments that have contributed for the CN research (Costa et al., 2007): (i) investigation of the random-network model (Erdos & Rényi, 1959; 1960); (ii) investigation of small-world networks (Watts & Strogatz, 1998); and (iii) investigation of scale-free networks (Barabási & Albert, 1999). Moreover, works from various fields of science have focused on the statistical analysis of such networks (Boccaletti, Latora, Moreno, Chavez, & Hwang, 2006; Costa et al., 2007; Dorogovtsev & Mendes, 2013; Newman, 2003).

Most of the works using CNs have two main steps: (i) model the problem as a network; and (ii) extract topological measures to characterize its structure. These features can be useful for discriminating different categories/classes, and, therefore, for creating techniques for pattern recognition. In this context, current works focus on exploring concise strategies for representing the CN according to the problem at hand.

Although the field of CNs is gaining a lot of attention from many areas, such as physics and biology, it is still an under-explored field in computer vision. Only a few works can be found in the literature which uses CNs as a supporting method. In computer vision, examples include texture analysis (Chalumeau, Meriaudeau, Laligant, & Costa, 2008; Gonçalves, Machado, & Bruno, 2015; Scabini, Gonçalves, & Castro Jr, 2015), nanoparticle agglomeration analysis (Machado et al., 2017), face recognition (Gonçalves, de Jonathan de Andrade Silva, & Bruno, 2010), and shape analysis (Backes et al., 2009); this last work presented good results, but its simplicity renders for limited discrimination, as we demonstrate in our comparative experiments.

2.1. Complex network representation and measures

As previously discussed, graphs are used to represent CNs. Specifically, in our work, we use undirected weighted graphs. In this representation, a graph $G = \{V, E\}$ accounts for $V = \{v_1, \ldots, v_n\}$, a set of n vertices; $E = \{e = (v_i, v_j) | v_i \in V \text{ and } v_j \in V\}$, a set of edges that connects pairs of vertices v_i and v_j; and values $e_{v_i,v_j} = weight(v_i, v_j)$ representing the edge weight of the connection between two vertices v_i and v_j.

There is a large number of measures that can be extracted from a CN, as presented in the work of Costa et al. (2007), in which the authors review different classes of measures. A simple but important one is the degree distribution. The degree of a vertex v_i is the number of its connections, which describes its interplay with neighbor vertices:

$$k(v_i) = \sum_{v_j} \begin{cases} 1, & \text{if } e_{v_i,v_j} \in E \\ 0, & \text{otherwise} \end{cases} \quad (1)$$

Most works use the degree distribution for the characterization of CNs, using measures such as the max and the average degree. However, although the degree is an important measure, using simple statistical measures from its distribution can limit the task of describing the properties of a network. For a more thorough analysis, other measures that consider the local properties of the vertices have the potential for a finer description.

2.2. Shapes as complex networks

The technique to model a shape as a CN was proposed in the work of Backes et al. (2009). Given a set of n_c pixels belonging to the contour of a shape in an image, a network $G = \{V, E\}$ is modeled so that each pixel is mapped to a vertex of the set $V = \{v_1, \ldots, v_{nc}\}$. The next step is the definition of the set of edges E. We connect each pair of vertices to define edges whose weights come from the Euclidean distance considering the position of the pixels. The distance between two pixels i and j represented by vertices v_i and v_j is given by:

$$d(v_i, v_j) = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \quad (2)$$

Next, we normalize the weights as follows:

$$e_{v_i,v_j} = \frac{d(v_i, v_j)}{d_{max}} \quad (3)$$

where $d_{\max}$ is the largest distance between all pixels. This normalization is performed so that the connections' weights are kept in the interval [0, 1]. It also keeps the edge's weights invariant to the scale of the shape.

At this point, the topology is regular – all the vertices have the same number of connections; a selective procedure is necessary, keeping only the edges that are more relevant. A simple and widely used approach is to remove edges according to a threshold. Given a threshold t, a new network $G^t = \{V, E^t\}$ is obtained by discarding edges that have a weight greater than t, as follows:

$$E^t = \{(v_i, v_j) | e_{v_i,v_j} \le t, v_i \in V, v_j \in V\} \quad (4)$$

The resulting G^t network fits into the Watts and Strogatz small-world model (Watts & Strogatz, 1998), and can be used as a representation of a given image's shape. Fig. 1 shows an example of a *shape network* after the threshold operation.

2.3. Dynamic analysis of complex networks

The threshold value t directly affects the network topology resulting in dense or sparse networks, as observed in Fig. 2. Moreover, a CN cannot be fully characterized without considering the interplay between structural and dynamic aspects (Costa et al., 2007). In our methodology, we access the dynamics of the network by using a set of thresholds $T = \{t_1, t_2, \ldots, t_n\}$. This analysis covers the network evolution since its creation (low thresholds) to its stabilization (high thresholds). The result is a shape represented by a

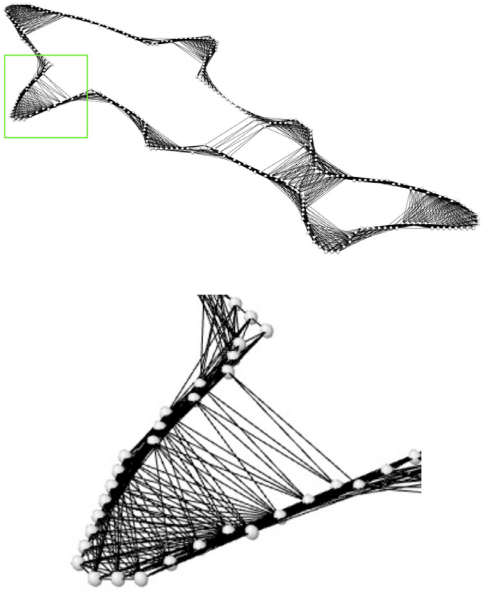


Fig. 1. A Watts and Strogatz small-world network built from a fish shape.

set of CNs, $\{G^{t_1}, G^{t_2}, \ldots, G^{t_n}\}$. The combination of the measures extracted from all the t networks results in a robust feature vector that characterizes the shape dynamics in terms of the neighborhood interplay of its pixels.

3. Proposed approach: angular descriptors of complex networks

This section presents our proposed methodology, which is based on the extraction of features from the topological transformations presented in Section 2. Our motivation was to improve the shape characterization by using local geometric measures instead of the global degree analysis observed in the work of Backes et al. (2009), discussed in the previous sections. The core of our method is to use the geometrical angles defined by the vertices in the shape contour, three at a time. We refer to the sets of features computed from the CN angles as Angular Descriptors of Complex Networks (ADCN).

In order to extract angular descriptors from CNs, the following steps are necessary. Given a network G^t, transformed by a threshold t, each vertex v_i is analyzed and the angle θ is computed between its incident edges with relation to two of its neighboring vertices v_j and v_k– see Fig. 3:

$$\theta_{v_i}^{v_j,v_k} = \frac{(\arctan(v_i - v_j) - \arctan(v_i - v_k)) * 180}{\pi} \tag{5}$$

Fig. 3 illustrates this approach for a vertex with three adjacent vertices. It is important to notice that we have normalized the angle interval into [0, 180] taking the inner angle between the edges ($\theta_{v_i}^{v_j,v_k} = \min(\theta_{v_i}^{v_j,v_k}, 360 - \theta_{v_i}^{v_j,v_k})$). An important issue is that the order of edges changes the resulting θ; therefore, we use a specific ordering of edges as part of the method, considering the edges with higher weight first. This process is discussed in depth in Section 3.2.

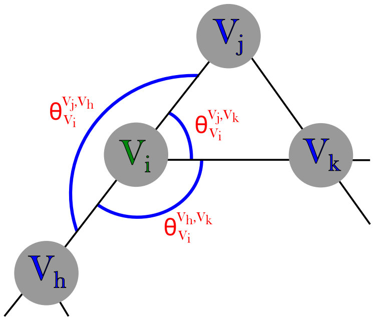


Fig. 3. A set of angles $[\theta_{v_i}^{v_j,v_h}, \theta_{v_i}^{v_h,v_k}, \theta_{v_i}^{v_k,v_j}]$ computed for a vertex v_i using its adjacent vertices v_j, v_k and v_h.

After calculating the angles for all the vertices of the network G^t, we compute a histogram ϕ^t of the frequency of the angles:

$$\phi^t = \left[\frac{h(0^\circ), h(1^\circ), h(2^\circ), \ldots, h(180^\circ)}{\sum_{k=0}^{180} h(k)}\right] \tag{6}$$

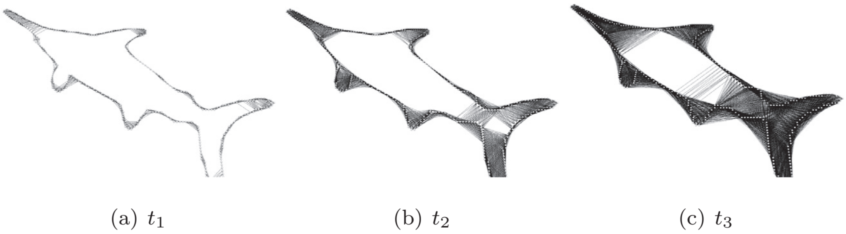
We normalize the histogram into a probability density function by dividing each element by the sum of its occurrences. This ensures invariance to linear transformations that might increase or decrease the number of occurrences. The histogram size was defined according to the probability of occurrence of each angle; the experiments demonstrated that each angle in the interval [0, 180] has a significant frequency.

Intuitively, the angle is a quite descriptive characteristic when it comes to shapes; in this aspect, the angular histogram is able to detect the corners, the abrupt changes, and the overall perimeter of a given shape. These properties are illustrated in Fig. 4, which compares the angular frequency of an airplane and of a leaf. As the leaf is almost circular, the occurrence of low angles is smaller if compared to the airplane, which has many sharp curves.

3.1. Feature vector

Using the angular histograms ϕ^t obtained with different t values, it is possible to construct a feature vector to characterize a given shape. First, it is important to notice that the use of the entire histograms results in a feature vector with a dimensionality that is too high; one single histogram has 181 bins, hence, the feature vector size would be n*181, which is impracticable. In the stead, we summarize the histograms by using statistical measures Mean, Standard deviation, Energy, Entropy, Contrast, and Homogeneity, see Table 1.

We have analyzed the combination of each measure, and also their individual performance; the combination of all the 6 measures presented the best performance. To define the final feature vector φ, we considered the dynamic analysis described in



(a) t_1 (b) t_2 (c) t_3

Fig. 2. Shape-network evolution through different thresholds $t_1 < t_2 < t_3$.

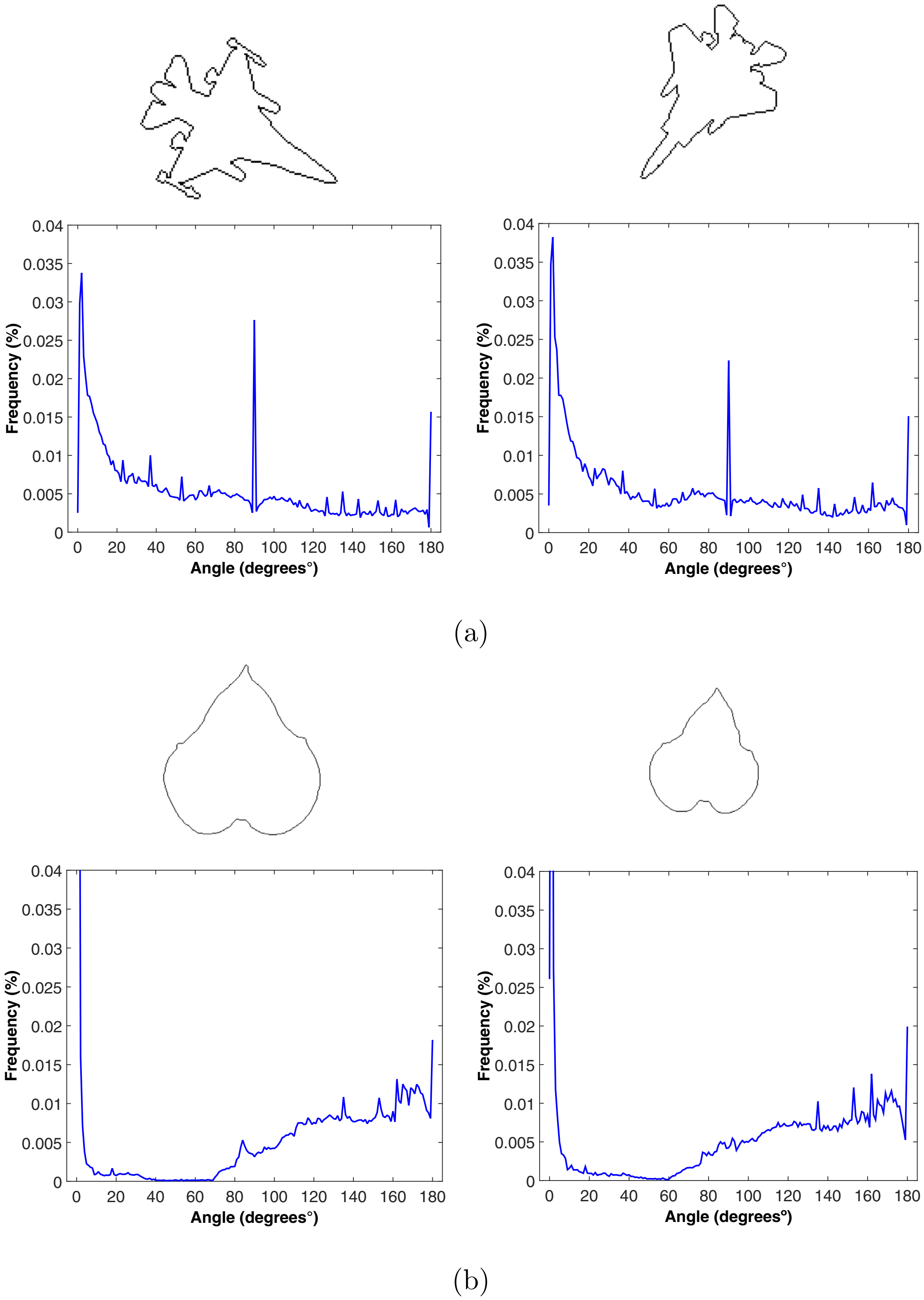


Fig. 4. Angular frequency extracted from shapes of two different classes. The network used for angular extraction was built with $t = 0.5$. (a) Airplane samples. (b) Leaf samples.

Section 2.3, i.e., we concatenated the 6 statistical measures extracted from networks with different thresholds t, as follows:

$$\varphi = [\mu^{t_1}, \sigma^{t_1}, e^{t_1}, \epsilon^{t_1}, C^{t_1}, H^{t_1}, \ldots, \mu^{t_n}, \sigma^{t_n}, e^{t_n}, \epsilon^{t_n}, C^{t_n}, H^{t_n}] \quad (7)$$

The entire process is illustrated in Fig. 5. First, the boundary pixels are obtained from the image, and then the CNs G^t are modeled for a set of thresholds $\{t_1, \ldots, t_n\}$. Each network is evaluated, providing a ϕ^t angle histogram, and the feature vector φ is obtained by the concatenation of its statistical measures.

Although the feature vector is composed of statistical measures, it is straight to notice that each individual vertex contributes to the process. For example, a vertex with a high degree will provide extensive information on its own; this is because the set of its incident edges will provide a broad set of angles. Such vertices are lo-

Table 1
Equations of the statistical summarizations used to compose the feature vector.

Mean	Standard deviation	Energy
$\mu^t = \sum_{i=0}^{180} \phi^t(i) * i$	$\sigma^t = \sqrt{\sum_{i=0}^{180} (i^2 \phi^t(i)) - \mu^2}$	$e^t = \sum_{i=0}^{180} \phi^t(i)^2$
Entropy	Contrast	Homogeneity
$\epsilon^t = -\sum_{i=0}^{180} \phi^t(i) * \log(\phi^t(i))$	$C^t = \sum_{i=0}^{180} \phi^t(i) * i^2$	$H^t = \sum_{i=0}^{180} \frac{\phi^t(i)}{i}$

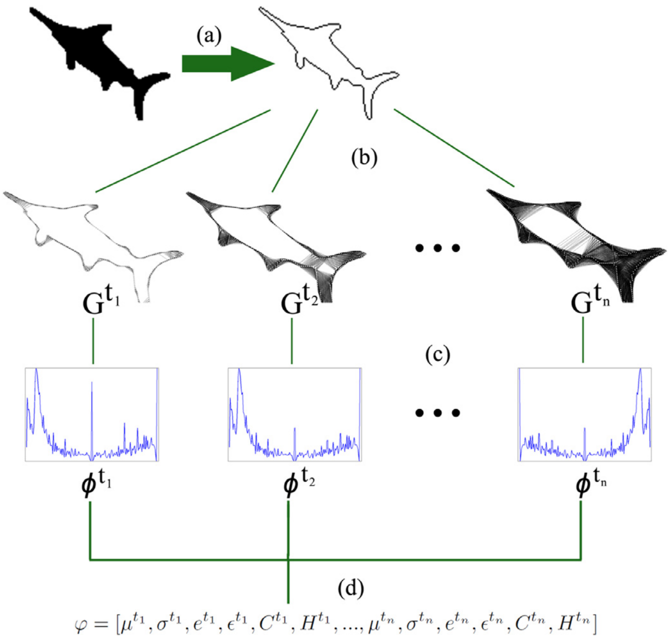

(a)
(b)
G^{t_1}
G^{t_2}
G^{t_n}
(c)
ϕ^{t_1}
ϕ^{t_2}
ϕ^{t_n}
(d)
$\varphi = [\mu^{t_1}, \sigma^{t_1}, e^{t_1}, \epsilon^{t_1}, C^{t_1}, H^{t_1}, ..., \mu^{t_n}, \sigma^{t_n}, e^{t_n}, \epsilon^{t_n}, C^{t_n}, H^{t_n}]$

Fig. 5. Steps to the extraction of angular descriptors from a shape image. (a) Contour pixels extraction. (b) CNs G^t for each threshold value. (c) Angular histogram of each CN. (d) Statistical measures computed from the angular histogram of each CN concatenated to compose the feature vector φ.

cated in regions with abrupt changes in the boundary of the shape, where rich information can be extracted for characterization.

3.2. Rotation and scale invariance

Rotation and scale invariance are essential in shape analysis since these transformations are very common in any application. To analyze the invariance of our approach, there are two aspects to be considered: (i) the network's topology; and (ii) the angular histogram (distribution). For both of these aspects, the normalization of the edges' weights (Section 2.2) plays an important role. As we use the distance between pixels and their coordinates to connect and define edges, rotation transformations have no influence on the network's topology. To analyze the network scale invariance, it is necessary to consider that images with different sizes, consequently, do not have the same number of contour pixels. Therefore, according to Section 2.1, two networks $G_1 = \{V_1, E_1\}$ and $G_2 = \{V_2, E_2\}$ constructed with images of different sizes do not satisfy to $|V_1| = |V_2|$; actually, they define networks with different number of vertices. However, since we consider the distribution of the angles and not their cardinality, it does not matter at which scale we are, for any given t the distribution is similar, as well as its statistical descriptors. We demonstrate this property in Fig. 6, which shows the angular histogram of the same shape, originally with size 128x128 pixels, after 4 scale operations. The similarity of the histograms is corroborated in Table 2 using the distance between them. The distance is computed by $\frac{1}{2}\sum|\phi_1(i) - \phi_2(i)|$, which means the average distance between the positions of the histograms, ranging from 0 to 180.

Table 2
Histogram distance to the original shape considering the histograms of four different scales; computations for two threshold values.

	+25%	+50%	+75%	+100%
$t = 0.33$	0.042	0.052	0.061	0.067
$t = 0.66$	0.053	0.078	0.088	0.098

Fig. 6 and Table 2 are emphatic in demonstrating scale invariance; for the same threshold value, the highest histogram distance is of 0.098, where values can reach up to 1. In Section 4.3, we go further into this discussion with experimental results.

3.3. Automatic threshold selection

One of the limitations faced in previous CN methods is the number of parameters needed to build the threshold set T, see Section 2.3. As previously discussed, this set is composed by an initial threshold t_1, a final threshold t_n and a set of thresholds in the range $[t_1, t_n]$. In previous works, the authors evaluated three parameters: the initial and final thresholds, and the increment factor. In the work of Backes et al. (2009), they empirically defined $t_1 = 0.025$, $t_n = 0.95$ and an increment factor of 0.075. However, manually defining a fixed threshold set is a cumbersome task that depends on trial and error.

Instead, we propose an approach that eliminates the task of choosing threshold values for each dataset, focusing on an automated method. It consists in modeling a Gaussian distribution for the edges of the training images. The first step of the method is to build the CN, $G_I = \{V_I, E_I\}$, of each image $I \in P$ in a training set P with $|P|$ images, obtaining the edges' weights of all the $|P|*|E|$ connections – no threshold cutting yet. Once the edges' weights are known, it is possible to estimate their mean and standard deviation for the training set P as follows:

$$\mu = \frac{1}{|P| * |E|} \sum_{I \in P} \sum_{(v_i, v_j) \in E_I} weight(v_i, v_j) \tag{8}$$

$$\sigma = \sqrt{\frac{1}{|P| * |E|} \sum_{I \in P} \sum_{(v_i, v_j) \in E_I} (weight(v_i, v_j) - \mu)^2} \tag{9}$$

Using these measures, it is possible to define a range that ensures a minimum coverage of the edges' weights distribution according to the three-sigma rule, as follows:

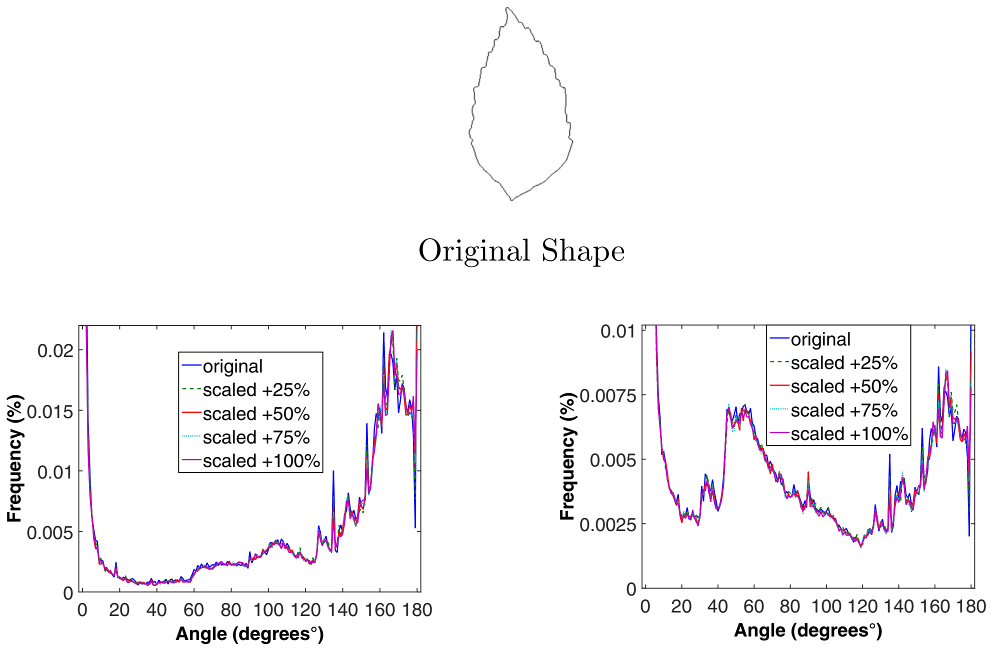
$$\begin{aligned} t_1 &= \mu - \alpha\sigma \\ t_n &= \mu + \alpha\sigma \end{aligned} \tag{10}$$

where α controls the covering rate of the distribution; for example, for $\alpha = 1$ the coverage is approximately 68.27% of the distribution, while $\alpha = 3$ covers approximately 99.73%.

With this approach, the threshold range is defined according to the type of shapes being analyzed, automatically adjusting for each set of images. Moreover, it reduces the number of parameters of the proposed approach to just the number of thresholds to be used, that is, the number of values between the interval $[t_1, t_n]$, which means that the user just needs to set the parameter n.

3.4. Complexity and processing time

In order to evaluate the use of angular descriptors in real-time applications, we performed a complexity and processing time analysis. Consider the number of contour pixels n_c of a given shape-image with $n_s = w * h$ pixels and a fully-connected network (i.e.,



(a) $t = 0.333$

(b) $t = 0.666$

Fig. 6. Angular histogram for 4 scales of the same shape and for two threshold values.

threshold $t = 1$, which is the “worst case”). First, to find and extract the contour pixels it is necessary to verify each pixel of the image, which results in a cost of $O(n_s)$. To model the CN, the cost is $O(n_c^2)$, i.e. the cost for evaluating the distance between each pair of contour pixels and connect them. To remove connections and perform the angle calculations, we have a cost of $O(n_c^2 \log n_c)$ to visit and sort each edge going through each vertex (we used priority queues to compose the edges set of each vertex). Therefore, the total complexity becomes $O(n_s + 2n_c^2 \log n_c)$, and by removing the lowest order terms, we get $O(n_s + n_c^2 \log n_c)$.

To estimate the processing time of the proposed approach, we used an AMD FX(tm)-4300 Quad-Core processor with 3.80 GHz of clock and 4GB of RAM memory, in a 64-bit operational system. The average mean time of 10 executions to extract the angular descriptors for a shape with 435 contour pixels in an image with 114 × 114 pixels (sample from the Generic Shapes dataset) was 0.08 seconds, which implies that ADCN has competitive running time for real-time applications.

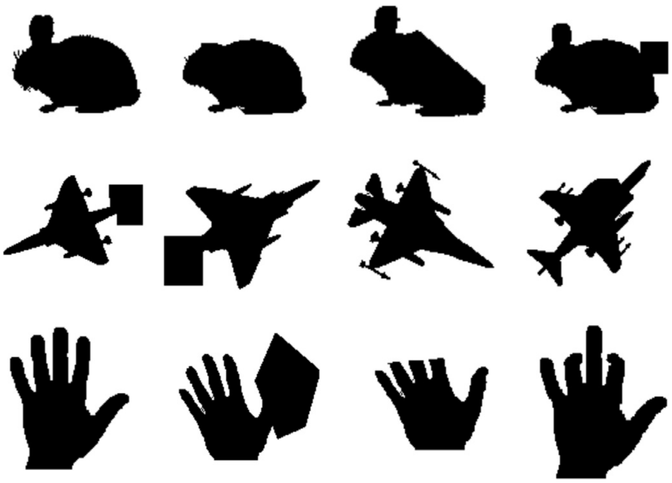


Fig. 7. Samples from the Generic shapes dataset; each row represents 4 samples from the same class.

4. Results and discussion

For experimentation, we used the same datasets and experimental protocol used by Backes et al. (2009), which consists of a classification step using Linear Discriminant Analysis (LDA) and of a Leave-one-out cross-validation. We present the parameter analysis of our approach, and also a comparison with other shape descriptors including the two original CN descriptors. Moreover, an analysis of noise and degradation robustness is performed in Sections 5.1 and 5.2.

4.1. Experimental design

4.1.1. Datasets

The analysis performed in this section includes classification results obtained by using the following datasets:

- Generic Shapes: also known as 99 Shapes, this dataset (Sebastian, Klein, & Kimia, 2004; Sharvit, Chan, Tek, & Kimia, 1998) is widely used in the literature; it is composed of 9 classes with 11 samples each, including transformations such as overlap, occlusion, missing parts and variations in structure; the full dataset has 99 images of varied sizes; samples from the Generic shapes dataset can be observed in Fig. 7.
- Fish: this is a large dataset with 11,000 images, composed by contours of fish species (Backes et al., 2009); it has a total of 1100 classes, with 10 samples each; its Samples present 10 different transformations, that is 5 rotations and 5 scales; some classes and samples are illustrated in Fig. 8.
- Leaves: this dataset is composed of natural leaves extracted from different plant species grouped into 30 classes, each one with 20 samples of sizes 256 × 256 (Backes et al., 2009); this is a very challenging dataset due to a high between-class similarity and due to within-class differences; moreover, overlaps and leaf deformation can occur in samples of the same class; the dataset is illustrated in Fig. 9.

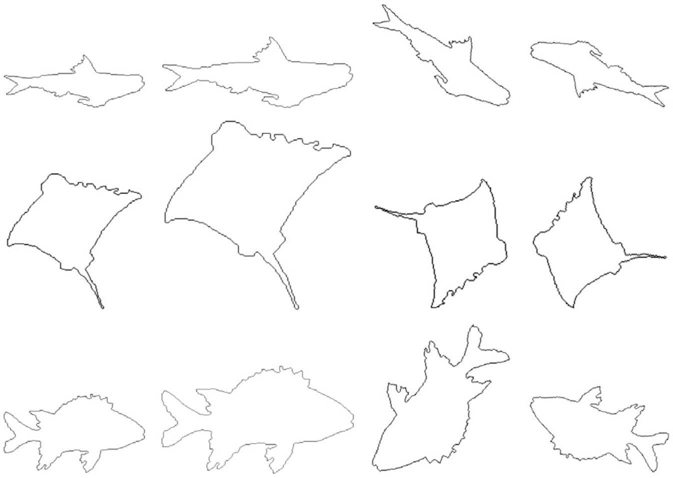


Fig. 8. Samples from the Fish dataset; each row represents 4 samples from the same class.

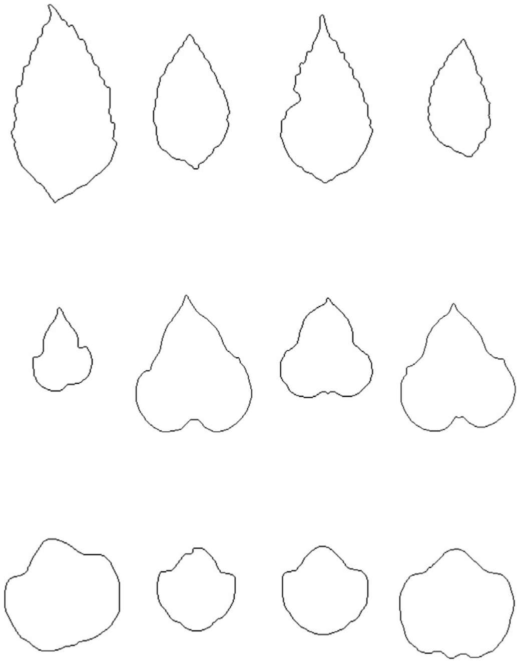


Fig. 9. Samples from the Leaves dataset; each row represents 4 samples from the same class.

- Rotated Leaves: for rotation invariance analysis, each sample of the original Leaves dataset was rotated according to the following angles: 7°, 35°, 104°, 132°, 201° and 298°; the new dataset consists of 30 classes with 120 samples each; Fig. 10 (a) illustrates the rotations that define the dataset.
- Scaled Leaves: for scale invariance analysis, a new dataset was obtained by scaling each sample of the original Leaves dataset by 25%, 50%, 75% and 100%, as seen in Fig. 10 (b); the new dataset is composed of 30 classes of 80 samples each.

These datasets have different properties, such as generic artificial shapes (Generic shapes) and real natural shapes (Fish and Leaves). This combination of datasets suits the analysis of the descriptors' robustness against many real issues and, also, their application in real problems.

4.1.2. Linear discriminant analysis

To evaluate the ADCN and other methods, we perform a classification task over its descriptors. We use the LDA classifier (Ripley, 1996), a well-known supervised method to estimate a linear subspace with good discriminating properties. The idea is to follow the same experimental protocol of the work of Backes et al. (2009) so to perform a fair comparison with the CN degree descriptors. The LDA consists of finding a projection of the features where the between-class variance is larger compared to the within-class variance. Nearest neighbor classification is performed to predict new samples. Using leave-one-out cross-validation, we measure the performance using the accuracy rate, that is the number of correctly classified samples predicted by LDA.

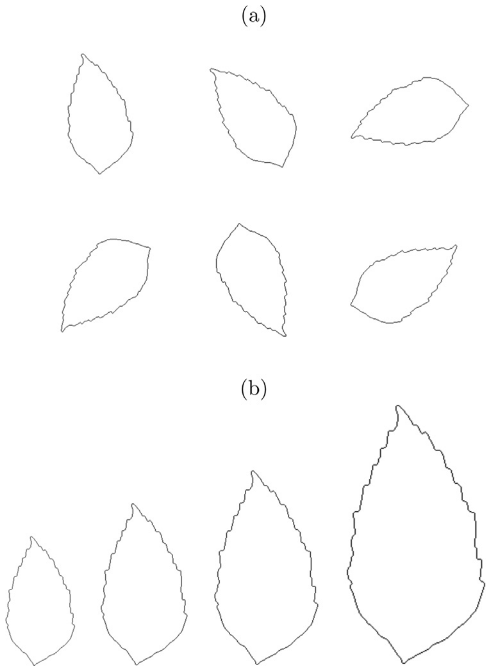


Fig. 10. Rotation (a) and scale (b) performed on a sample of the original Leaves dataset.

4.2. Parameter analysis

This section presents an analysis of the automatic threshold selection considering the number of thresholds n.

4.2.1. Threshold range

Considering the dynamic analysis of CN-based descriptors, as previously discussed in Section 2.3, it is necessary to define a set of thresholds $T = \{t_1, \ldots, t_n\}$ to obtain CNs $\{G^{t_1}, \ldots, G^{t_n}\}$, which provide measures to compose the feature vector.

To define the set T, first, it is necessary to define the initial and the final thresholds (t_1 and t_n). Instead of manually defining values, as was done in previous works, we used automatically-defined thresholds, as defined in Section 3.3. Therefore, only the number of thresholds n must be defined, so that the set T can be divided into n equidistant values ranging from t_1 to t_n. Accordingly, we evaluated Eq. (10) by varying α from 1 to 3 increasing it by 0.2. Fig. 11 (a) shows the accuracy rate in each dataset as we increase the value of α. From the figure, one can see that for $\alpha = 1.4$, we get the best accuracy rate. In Fig. 11 (b), we use this value of α varying the value of n, which we discuss in Section 4.2.2.

It was observed that a covering range using α between 1 and 1.4 presents the best results, except for the Fish dataset, which was not significantly sensitive to α. With $\alpha = 1.4$, the resulting threshold ranges for each dataset are: [0.1059, 0.7645] (Leaves), [0.1118, 0.7445] (Generic Shapes), and [0.0693, 0.7063] (Fish).

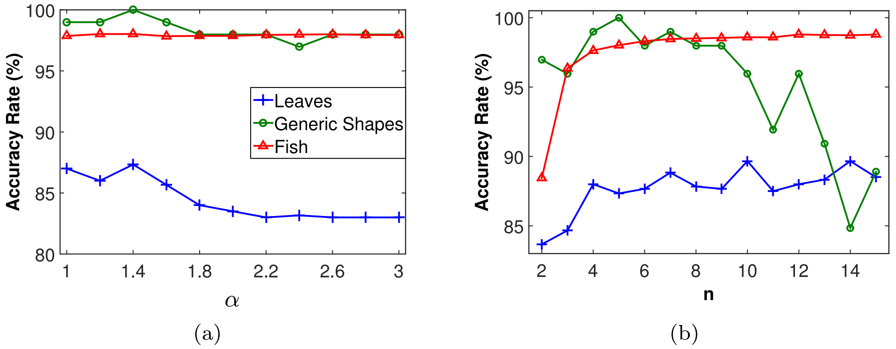


Fig. 11. Accuracy rate on each dataset; (a) by varying α (using $n = 5$); and (b) by varying n (using $\alpha = 1.4$).

To achieve a threshold range close to the fixed range used by Backes et al., it would be necessary to use an α value close to 2.4, which achieves an inferior performance, as the accuracy rates show in Fig. 11(a). Therefore, we can argue that applying low or high values like 0.025 and 0.955, as was done in the work of Backes et al., may include CNs with no relevant information (sparse or regular topologies).

4.2.2. Number of thresholds

To analyze the number of thresholds, we evaluated n ranging from 2 to 14 for $\alpha = 1.4$. The results can be observed in Fig. 11 (b), which presents the accuracy rates for each dataset. We stopped the analysis at $n = 14$ due to the drop in the accuracy rate of the Generic shapes dataset and due to the stable accuracy of the Fish and Leaves datasets. The best results are achieved with $n = 10$ and $n = 14$ in the Leaves dataset (89.67%); $n = 5$ in the Generic shapes dataset (100%); and $n = 12$ in the Fish dataset (98.79%). In order to find a balance between keeping a significant number of features and a good performance, and considering the number of features of other descriptors, we defined $n = 7$ as the best parameter, considering that the results are satisfactory for each dataset and the number of features is 42. We believe that it can be generally used and that it will present satisfactory results for a large range of image domains.

4.3. Comparison with other shape descriptors

To compare our approach with the state of the art, we evaluated the following shape descriptors considering their proposed implementation and parameters:

- CN degree and joint degree descriptors (Backes et al., 2009): the degree descriptors concatenate the max and the average of the degree distribution of the CNs to compose the feature vector; the joint degree descriptors consist of the distribution of measures entropy, energy, and average computed from the joint degree of the network. We used the same parameters defined by the authors in their original work, that is, a fixed threshold set ranging from $T_{ini} = 0.025$ incremented at a regular interval of 0.075 until reaching a final threshold $T_Q = 0.95$.
- Fourier descriptors (Osowski et al., 2002): this descriptor is composed of a feature vector with the 20 most significant coefficients of the Fourier transform of the contour.
- Zernike moments (Zhenjiang, 2000): this approach represents the shape by a feature vector containing 20 moments (the order ranging from $n = 0$ to $n = 7$) corresponding to the most significant magnitudes of a set of orthogonal complex moments of the image.
- Curvature descriptors (Wu & Wang, 1993): the shape is represented as a curve, where its maximum and minimum local points correspond to the direction changes in the shape contour.
- Multiscale fractal dimension (Plotze et al., 2005; Torres et al., 2004): using concepts from the fractal theory, the shape is represented as a curve that describes how the contour complexity changes as the scale changes. To compose the feature vector, it considers the 50 most meaningful points of the curve.

Table 3
Number of features used by each descriptor.

Descriptor	n	no. of features
ADCN	5	30
ADCN	7	42
ADCN	10	60
CN degree		26
CN joint degree		39
Fourier		20
Zernike		20
Curvature		25
Multiscale fractal dimension		50

Table 4
Classification results achieved by each descriptor in the Generic shapes dataset.

Descriptor	Qty correctly classified	Accuracy rate (%)
ADCN ($n = 7$)	98	98.99
ADCN ($n = 5$)	99	100
CN degree	95	95.96
CN joint degree	86	86.87
Fourier	83	83.84
Zernike	91	91.92
Curvature	76	76.77
Multiscale fractal dimension	87	87.88

Table 3 presents the number of features used by each descriptor. Our method, the Angular Descriptors of Complex Networks (ADCN), uses 6*n features, which correspond to the concatenation of the 6 distribution measures, as described in Table 1, extracted for each of the n thresholds. In the following results, we show the accuracy rates achieved using the standard parameter ($n = 7$) and also the value n that provides the best results.

4.3.1. Generic shapes dataset

First, we evaluated the classification results of each descriptor in the Generic shapes dataset. As previously discussed, this dataset presents many kinds of transformations, such as overlap, variations in structure, and missing parts. Despite the challenges, the proposed method presents promising results. Table 4 shows the results achieved by each descriptor. The ADCN presents the best performance, with 98.99% against 95.96% of the CN degree descriptors. It is also important to note that the result of 100% achieved with $n = 5$ is the highest so far reported in the literature for this spe-

Table 5
Classification results achieved by each descriptor in the Fish dataset.

Descriptor	Qty correctly classified	Accuracy rate (%)
ADCN ($n = 7$)	10,831	98.46
ADCN ($n = 12$)	10,867	98.79
CN degree	10,932	99.38
CN joint degree	10,428	94.80
Fourier	10,897	99.07
Zernike	1345	12.23
Curvature	10,730	97.55
Multiscale fractal dimension	4105	37.32

Table 6
Accuracy rate of each descriptor on the leaves dataset.

Type of experiment	Descriptor	Qty correctly classified	Accuracy rate (%)
Original 600 images	ADCN ($n = 7$)	534	88.83
	ADCN ($n = 10$)	538	89.67
	CN degree	502	83.67
	CN joint degree	461	76.83
	Fourier	450	75.00
	Zernike	408	68.00
	Curvature	450	75.00
	MS fractal dimension	438	73.00
Rotated 3600 images	ADCN ($n = 7$)	3188	88.56
	ADCN ($n = 10$)	3239	89.97
	CN degree	3020	83.89
	CN joint degree	2992	78.22
	Fourier	2755	76.53
	Zernike	2517	69.92
	Curvature	2831	78.64
	MS fractal dimension	2455	68.19
Scaled 2400 images	ADCN ($n = 7$)	2145	89.38
	ADCN ($n = 10$)	2161	90.04
	CN degree	2019	84.12
	CN joint degree	2007	79.45
	Fourier	1958	81.58
	Zernike	1309	54.54
	Curvature	1920	80.00
	MS fractal dimension	1784	74.33

cific dataset. Considering these results, the promising performance of the angular descriptors is evident, overcoming all the other descriptors. This implies that it has the potential to handle deformations in the shapes and to generalize in cases of classes with low within-class similarity.

4.3.2. Fish dataset

The experimentation over the Fish dataset allowed for the evaluation of performance in a large scale. This dataset has 11,000 images and 1100 classes. Table 5 shows the accuracy rate of each descriptor. The ADCN reached 98.79%, against 99.38% achieved by the CN degree descriptors, and 99.07% of the Fourier descriptors. Although the previous method surpassed our methodology for this specific case, our results are still competitively effective, presenting a maximum difference of only 0.59% for the accuracy rate. On the other hand, the methods based on Zernike moments and on Multiscale fractal dimension drastically lost performance compared to their results over the Generic shapes dataset.

4.3.3. Leaves dataset

Finally, we experiment on the Leaves dataset and its variations. Table 6 presents the accuracy rate and the number of correctly classified images for the original, scaled, and rotated versions of this dataset. In the original dataset, the ADCN reached 89.67% of accuracy rate using $n = 10$, and 88.83% with $n = 7$, followed by the CN degree descriptors with 83.67%. The other descriptors achieved results below 75%. This emphasizes the challenges faced in the classification of leaves, which requires more advanced techniques.

Table 7
The accuracy rate of each descriptor for 600 images of the Leaves dataset at each noise level. For this experiment, we used $n = 7$ as the parameter to the proposed method ADCN.

Descriptor	Accuracy rate (%)			
	Level 1	Level 2	Level 3	Level 4
ADCN	83.33	81.67	81.00	80.83
CN degree	83.00	81.83	80.50	80.12
CN joint degree	77.67	76.17	75.17	75.83
Fourier	69.17	64.00	62.83	57.67
Zernike	66.50	67.00	65.67	66.50
Curvature	75.33	74.83	72.00	71.33
MS fractal dimension	64.00	57.50	56.67	52.17

The results achieved in the rotated version of the Leaves dataset is an evidence that the angular descriptors method is invariant to rotations. The proposed method maintains its performance, providing 89.97% with $n = 10$, and 88.56% with $n = 7$, against 83.89% from the CN degree descriptors. In general, the descriptors maintained their performance considering the results achieved in the original dataset, except for the MS fractal dimension descriptor that had its accuracy rate decreased from 73% to 68.19%.

The results over the Fish dataset also indicate robustness to scale, as previously discussed in Section 3.2. The ADCN reached 90.04% of accuracy rate with $n = 10$ and 89.38% with $n = 7$, overcoming the other descriptors. The CN degree descriptors reached 84.12%, followed by the Fourier descriptors with 81.58%. The moments of Zernike is not as tolerant to scale changes as other descriptors, and its accuracy rate has remarkably dropped.

5. Noise and degradation aspects

We also evaluated the descriptors performance in other aspects such as noise and degradation, considering that in real applications the shape can be affected by these kinds of deformations. In this context, to complement the analysis of the proposed descriptors, we also perform experiments adding 4 noise levels and 2 types of degradations to the images. Fig. 12 shows samples of the noise and degradation used in the experiments.

5.1. Robustness against noise

In this section, we present classification results in noisy images in order to complement the analysis of the descriptors' robustness, evaluating their tolerance to noise. To perform the experiment, the same dataset used in the work of Backes *et al.* was used. It is composed by noise transformations applied to the original samples of the Leaves dataset. The noise is uniformly generated at the interval $[-r, \ldots, r]$, where r is the intensity level of the noise, and then added to the original signal. The noise pattern is generated for both the x and y coordinates of the contour. Each noise level results in a dataset with 600 images. The results achieved by the descriptors for each noise level are presented in Table 7.

As expected, the ADCN lost performance in comparison to the results achieved in the original dataset. This behavior was expected due to the noise influence in the angle computations, which is the main property of the angular descriptors. Nevertheless, the proposed approach presents superior or equivalent results if compared with other descriptors; it presented its worst performance at noise level 2, but still, it was superior to the competitors.

Another kind of problem is the existence of noise that is not part of the contour pixels. We want to raise some points concerning this issue related to our method. In principle, this noise will contribute to the weight of the edges, as we consider the largest Euclidean distance between the contour pixels and its normalization. Thus, the weight of the edges between the contour pixels

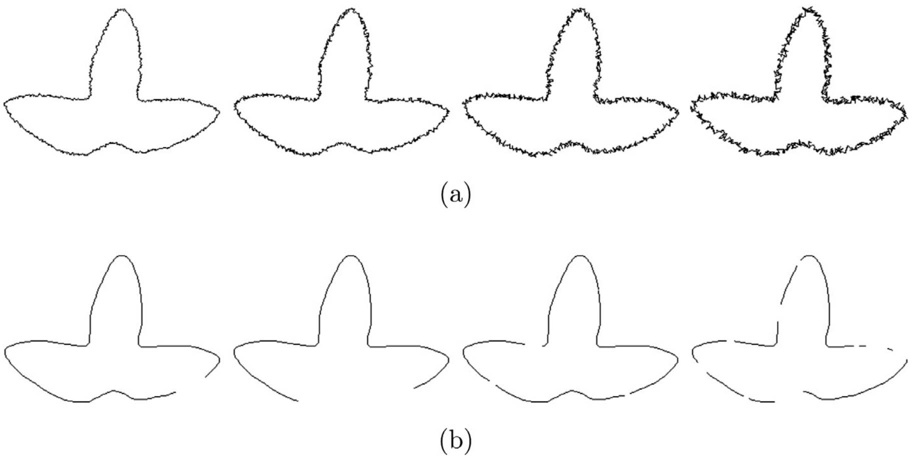


Fig. 12. Samples of (a) 4 noise levels; and (b) 4 degradations (2 continuous and 2 random).

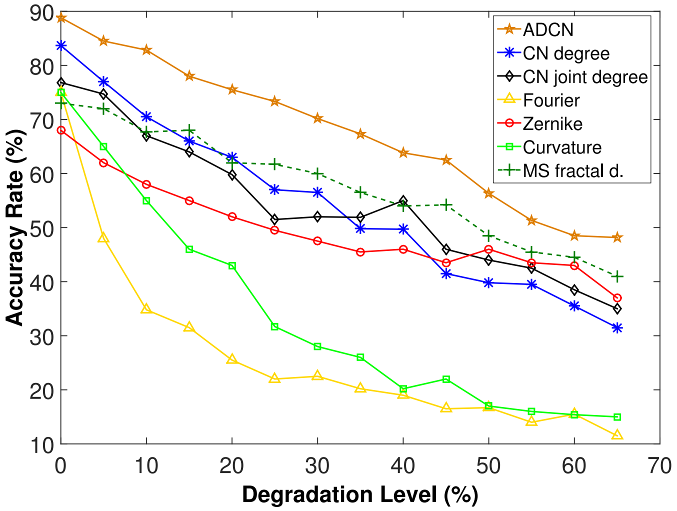


Fig. 13. Behavior of shape descriptors in the Leaves dataset with several levels of continuous degradation.

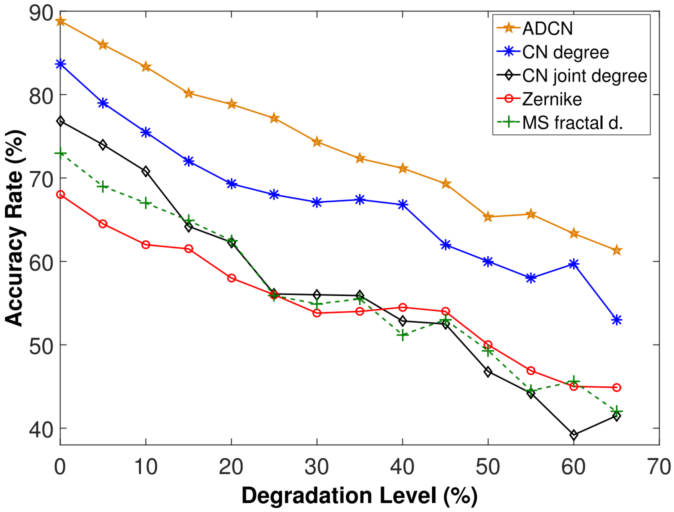


Fig. 14. Behavior of shape descriptors in the Leaves dataset with several levels of random degradation.

and the noise tends to be higher (close to 1). However, considering our proposal of automatically defining the threshold, we obtain the distribution of the edges' weights from all the training images; that is, connections between contour pixels will occur much more often than to noise pixels. We then select threshold values around the mean based on the Gaussian parameters of the distribution. It means that we remove extreme values (tails of the distribution), which then imply in the removal of high weight connections, like those related to noise that is not part of the contour.

5.2. Robustness against degradation

The proposed descriptor is also evaluated in a dataset with degradation on the shapes. For this analysis, several levels of degradation are applied to the original samples of the Leaves dataset. First, we evaluate long continuous discontinuities found in the shape's contour. We measure the descriptor behavior as the level of degradation is increased, applying 13 degradation levels (5%, 10%, ... , 65%). Fig. 13 shows the accuracy rate for each level of degradation applied to the Leaves dataset.

We also evaluated random degradation, that is, short random discontinuities found in the shape's contour. In this experiment, Fourier and Curvature descriptors cannot be evaluated because they do not work over segmented contours. The levels of random degradation follow the same factor as those applied in the continuous degradation (5%, 10%, ... , 65%). Fig. 14 shows the performance of each shape descriptor over the random degradations of the Leaves dataset.

According to these results, it is valid to assume that the ADCN has great robustness against degradation levels. The curves presented in the continuous and random experiments indicate that the proposed approach keeps its performance above the other descriptors for each evaluated level of degradation. The CN degree descriptors presented good performance in the random degradation; on the other hand, it is not as tolerant as other descriptors in the continuous degradation, such as the MS fractal dimension and the Zernike moments. The Fourier and Curvature descriptors proved to be highly influenced by degradation, considerably losing accuracy rate.

In the same image, degradation can often occur simultaneously with another type of noise. As previously discussed, the noise around the contour has a high influence in the angular calculations, which results in performance loss. On the other hand, experiments demonstrated that our method is robust to some degradation types and levels, compared to other methods. With these findings, we believe that if both noise (around the contour) and degradation are present, the performance of the angular descriptors will be mostly influenced by noise, achieving similar results as compared to those presented in our noise experiments.

6. Conclusion

We introduced a novel method to extract features to represent shapes found in digital images. Given an image shape, the method builds a CN over the boundaries of the shape – pixels over the boundaries correspond to vertices, and the distances among them define weighted edges according to a threshold; after the network is built, the next step is to compute the angles defined by the edges of the network; finally, the distribution of the angles is computed to create a histogram whose statistical measures are used to construct feature vectors named *Angular Descriptors* (ADCN). We demonstrated the efficacy of our method over a classification task carried out over five robust image datasets. The classification results demonstrated the method to be superior to previous works based only on the vertex degree feature of the shapes' CN, and, also, superior to methods based on Zernike moments, Multiscale Fractal dimension, and Fourier transform. We also demonstrated that the method is invariant to scale, and to rotation, and that it performs well even when image degradation occurs.

With these results, the use of CNs proved to be a promising approach for pattern recognition. We expect that future research is done to explore additional CN features in applications related to image retrieval, classification, and computer vision. A possible line of work is to use CNs for region-based shape analysis, considering all the pixels of the shapes for building the CNs. Moreover, as discussed in our experiments, even with the competitive performance achieved on the noisy images, the ADCN accuracy dropped considerably compared to the original images. In this context, alternatives shall be investigated to improve the noise robustness of the method. This methodology can also be used to extract measures from sub-complex networks (i.e., subgraphs). Thus, one might find multiple shapes within a single CN through sub-complex network matching.

Acknowledgments

The authors acknowledge Fundect (Grant number 071/2015), CNPq (444985/2014-0 and CNPq 134558/2016-2), Fapesp (2016/02557-0) and the programs PET - Fronteira and NERDS da Fronteira to the financial support that allowed the development of this research. We also acknowledge Andre R. Backes for providing the datasets for the experiments.

References

Backes, A. R., Casanova, D., & Bruno, O. M. (2009). A complex network-based approach for boundary shape analysis. *Pattern Recognition, 42*(1), 54–67.

Barabási, A.-L., & Albert, R. (1999). Emergence of scaling in random networks. *Science, 286*(5439), 509–512.

Boccaletti, S., Latora, V., Moreno, Y., Chavez, M., & Hwang, D.-U. (2006). Complex networks: Structure and dynamics. *Physics Reports, 424*(4), 175–308.

Chalumeau, T., Meriaudeau, F., Laligant, O., & Costa, L. d. F. (2008). Complex networks: application for texture characterization and classification. In *Elcvia: Electronic letters on computer vision and image analysis: 7* (pp. 093–100).

Chen, C.-C., & Tsai, T.-I. (1993). Improved moment invariants for shape discrimination. In *San Diego'92* (pp. 270–280). International Society for Optics and Photonics.

Costa, L. d. F., Oliveira Jr, O. N., Travieso, G., Rodrigues, F. A., Villas Boas, P. R., Antiqueira, L., et al. (2011). Analyzing and modeling real-world phenomena with complex networks: A survey of applications. *Advances in Physics, 60*(3), 329–412.

Costa, L. d. F., Rodrigues, F. A., Travieso, G., & Villas Boas, P. R. (2007). Characterization of complex networks: A survey of measurements. *Advances in Physics, 56*(1), 167–242.

Dorogovtsev, S. N., & Mendes, J. F. (2013). *Evolution of networks: From biological nets to the internet and WWW*. Oxford University Press.

Erdos, P., & Rényi, A. (1959). On random graphs i. *Publ. Math. Debrecen, 6*, 290–297.

Erdos, P., & Rényi, A. (1960). On the evolution of random graphs. *Publication of the Mathematical Institute of the Hungarian Academy of Sciences, 5*, 17–61.

Gonçalves, W. N., Machado, B. B., & Bruno, O. M. (2015). A complex network approach for dynamic texture recognition. *Neurocomputing*, 211–220.

Gonçalves, W. N., de Jonathan de Andrade Silva, & Bruno, O. M. (2010). A rotation invariant face recognition method based on complex network. In *Progress in pattern recognition, image analysis, computer vision, and applications*. In *Lecture Notes in Computer Science: 6419* (pp. 426–433). Springer.

Kim, W.-Y., & Kim, Y.-S. (2000). A region-based shape descriptor using zernike moments. *Signal Processing: Image Communication, 16*(12), 95–102.

Loncaric, S. (1998). A survey of shape analysis techniques. *Pattern Recognition, 31*(8), 983–1001.

Machado, B. B., Scabini, L. F., Orue, J. P. M., de Arruda, M. S., Goncalves, D. N., Goncalves, W. N., et al. (2017). A complex network approach for nanoparticle agglomeration analysis in nanoscale images. *Journal of Nanoparticle Research, 19*(2), 65. doi:10.1007/s11051-017-3760-7.

Mehtre, B. M., Kankanhalli, M. S., & Lee, W. F. (1997). Shape measures for content based image retrieval: A comparison. *Information Processing and Management, 33*(3), 319–337.

Mouine, S., Yahiaoui, I., & Verroust-Blondet, A. (2013). A shape-based approach for leaf classification using multiscaletriangular representation. In *Proceedings of the 3rd acm conference on international conference on multimedia retrieval* (pp. 127–134). ACM.

Neto, J. C., Meyer, G. E., Jones, D. D., & Samal, A. K. (2006). Plant species identification using elliptic fourier leaf shape analysis. *Computers and Electronics in Agriculture, 50*(2), 121–134.

Newman, M. E. (2003). The structure and function of complex networks. *SIAM Review, 45*(2), 167–256.

Osowski, S., et al. (2002). Fourier and wavelet descriptors for shape recognition using neural networks—a comparative study. *Pattern Recognition, 35*(9), 1949–1957.

Persoon, E., & Fu, K.-S. (1977). Shape discrimination using fourier descriptors. *IEEE Transactions on Systems, Man, and Cybernetics, SMC-7*(3), 170–179. doi:10.1109/TSMC.1977.4309681. Cited By 578.

Plotze, R. d. O., Falvo, M., Pádua, J. G., Bernacci, L. C., Vieira, M. L. C., Oliveira, G. C. X., et al. (2005). Leaf shape analysis using the multiscale minkowski fractal dimension, a new morphometric method: A study with passiflora (passifloraceae). *Canadian Journal of Botany, 83*(3), 287–301.

Ripley, B. D. (1996). *Pattern recognition and neural networks*. Cambridge University Press.

Scabini, L. F. S., Gonçalves, W. N., & Castro Jr, A. A. (2015). Texture analysis by bag-of-visual-words of complex networks. In *Progress in pattern recognition, image analysis, computer vision, and applications* (pp. 485–492). Springer.

Sebastian, T. B., Klein, P. N., & Kimia, B. B. (2004). Recognition of shapes by editing their shock graphs. *Pattern Analysis and Machine Intelligence, IEEE Transactions on, 26*(5), 550–571.

Sharvit, D., Chan, J., Tek, H., & Kimia, B. B. (1998). Symmetry-based indexing of image databases. In *Content-based access of image and video libraries, 1998. Proceedings. Ieee workshop on* (pp. 56–62). IEEE.

Shen, L., Rangayyan, R. M., & Desautels, J. L. (1994). Application of shape analysis to mammographic calcifications. *Medical Imaging, IEEE Transactions on, 13*(2), 263–274.

Torres, R. d. S., Falcão, A. X., & Costa, L. d. F. (2004). A graph-based approach for multiscale shape analysis. *Pattern Recognition, 37*(6), 1163–1174.

Wang, L., Tan, T., Hu, W., & Ning, H. (2003). Automatic gait recognition based on statistical shape analysis. *Image Processing, IEEE Transactions on, 12*(9), 1120–1131.

Watts, D. J., & Strogatz, S. H. (1998). Collective dynamics of 'small-world' networks. *Nature, 393*(6684), 440–442.

Wu, W.-Y., & Wang, M.-J. J. (1993). Detecting the dominant points by the curvature-based polygonal approximation. *CVGIP: Graphical Models and Image Processing, 55*(2), 79–88.

Xiao, X.-Y., Hu, R., Zhang, S.-W., & Wang, X.-F. (2010). Hog-based approach for leaf classification. In *Advanced intelligent computing theories and applications. with aspects of artificial intelligence* (pp. 149–155). Springer.

Zhang, D., & Lu, G. (2004). Review of shape representation and description techniques. *Pattern Recognition, 37*(1), 1–19.

Zhenjiang, M. (2000). Zernike moment-based image shape analysis and its application. *Pattern Recognition Letters, 21*(2), 169–177.

Leonardo Scabini is a master student in applied physics at Universidade de São Paulo, Brazil, and received his B.S. degree in computer science at the Federal University of Mato Grosso do Sul, Brazil, 2015. His research interests include computer vision and artificial intelligence, approaching topics such as feature extraction, image classification, complex networks and computational physics.

Danilo de Oliveira Fistarol is an undergraduate student in computer science at the Federal University of Mato Grosso do Sul, Brazil. His current research interests include texture analysis, image processing and computer vision.

Sávio Vinícius Albieri Barone Cantero is an undergraduate student in computer science at the Federal University of Mato Grosso do Sul, Brazil. His current research interests include texture analysis, image processing and computer vision.

Wesley Nunes Gonçalves received the B.S. degree in computer engineering from Dom Bosco Catholic University, Brazil, in 2007, the M.S. degree in computer science from University of São Paulo, Brazil, in 2010, and the Ph.D. degree in computational physics from University of São Paulo, Brazil, in 2013. He is a Professor at Federal University of Mato Grosso do Sul, Brazil, and is the author of several refereed papers in International Journals or Conference Proceedings. His topics of research include texture analysis, computer vision, complex networks, and the application of computational methods in precision agriculture.

Bruno Brandoli Machado is a Professor at Federal University of Mato Grosso do Sul, Brazil. He received his M.Sc. and Ph.D. in Computer Science from University of São Paulo, the last in 2016. His main research interests are image analysis and computer vision with focus on texture analysis, complex networks and deep learning methods. His research has lately been focused on the application of computational methods in precision agriculture. Professor Bruno is currently taking part in the Research Network of Agritechnology between Brazil and the UK.

José Fernando Rodrigues Júnior is a Professor at University of Sao Paulo, Brazil. He received his Ph.D. from this same university, part of which was carried out at Carnegie Mellon University, USA, in 2007. Rodrigues is a regular reviewer and author in his field, having contributed with publications in major journals and conferences. His topics of research include data analysis, content-based data retrieval, visualization, and the application of such techniques in the analysis of e-learning domains.