



A complex network approach for dynamic texture recognition

Wesley Nunes Gonçalves^{a,*}, Bruno Brandoli Machado^{b,c}, Odemir Martinez Bruno^d

^a Federal University of Mato Grosso do Sul, Rua Itibiré Vieira, s/n, Ponta Porã - MS CEP 79907-414, Brazil

^b University of São Paulo, Institute of Mathematics and Computer Science, Scientific Computing Group, Brazil

^c Federal University of Mato Grosso do Sul, Rua Itibiré Vieira, s/n, Ponta Porã - MS CEP 79907-414, Brazil

^d University of São Paulo, São Carlos Institute of Physics, Scientific Computing Group, Brazil

ARTICLE INFO

Article history:

Received 13 March 2014

Received in revised form

31 October 2014

Accepted 13 November 2014

Communicated by Jungong Han

Available online 24 November 2014

Keywords:

Dynamic textures

Complex networks

Texture analysis

ABSTRACT

In this paper, we propose a novel approach for dynamic texture representation based on complex networks. In the proposed approach, each pixel of the video is mapped into a node of the complex network. Initially, a regular complex network is obtained by connecting two nodes if the Euclidean distance between their related pixels is equal or less than a given radius. For each connection, a weight is defined by the difference of the pixel intensities. Given the regular complex network, a function is applied to remove connections whose weight is equal to or below a given threshold. Finally, a feature vector is obtained by calculating the spatial and temporal average degree for networks transformed by different values of threshold and radius. The number of connections of pixels from the same frame and from different frames, respectively, gives the spatial and temporal degrees. Experimental results using synthetic and real dynamic textures have demonstrated the effectiveness of the proposed approach.

© 2014 Elsevier B.V. All rights reserved.

1. Introduction

Dynamic textures have emerged as new field of investigation that can be defined as visual phenomena that exhibit spatial and temporal regularity. This recent field of investigation extends the texture image concepts to the spatio-temporal domain. Dynamic textures present the following characteristics [1]: (i) richer content compared to individual images; (ii) spatial and temporal regularity; (iii) large amount of raw data; and (iv) very little prior structure. These characteristics appear in a wide range of videos, which make their analysis important in several applications of computer vision. Examples of applications include classification of corrosion evolution of different nanostructures, classification of the growth of bacterial populations, monitoring traffic or crowds, detection of disasters such as fires.

The dynamic texture methods aim at characterizing the video in terms of its content, so that, it is possible to classify the videos into categories. Most of the existing methods can be divided into four categories: (i) motion-based methods, (ii) spatio-temporal filtering and transform-based methods, (iii) model-based methods, and (iv) spatio-temporal geometric property based methods. The methods from the motion-based category are the most popular due to the efficient estimation of measures from motion, such as optical flow

[2–4]. Motion is a natural way to describe the local dynamics and then stands for a powerful cue for visual recognition. The spatio-temporal filtering and transform-based methods describe dynamic textures at different scales in space and time through spatio-temporal filters, such as wavelet transform [5–7]. The model-based methods represent dynamic textures by means of mathematical models [8,9], such as the generative models (e.g. linear dynamical systems – LDS) [8,10–13]. These generative models naturally provide a representation that can be used in applications of synthesis [14], segmentation [12,8] and classification [11,15,12]. Recently in this category, agent methods have been proposed to dynamic texture analysis [9]. These methods use deterministic partially self-avoiding walks to extract appearance and temporal features from the dynamic textures. Finally, the spatio-temporal geometric property based methods are based on properties of moving contour surfaces [16]. From the trajectory surfaces, it is possible to extract motion and appearance features based on the tangent plane distribution [16]. Although major advances have been reported, the dynamic texture representation still remains an open and difficult problem due to the characteristics of dynamic textures. The reader may consult [17] for a review of dynamic texture methods.

In this work, we propose a new approach for dynamic texture recognition based on complex networks. The complex network popularity relies on the flexibility and generality for representing any natural structure, including dynamic textures. As far as the authors know, this work is the first to report the use of complex networks on dynamic texture description. In the proposed approach,

* Corresponding author.

E-mail addresses: wesley.goncalves@ufms.br (W.N. Gonçalves),

bruno.brandoli@ufms.br (B.B. Machado), bruno@ifsc.usp.br (O.M. Bruno).

URLs: <http://scg.ifsc.usp.br> (B.B. Machado), <http://scg.ifsc.usp.br> (O.M. Bruno).

a complex network represents the video of the dynamic texture by mapping each pixel into a node and connecting two nodes if the distance of their related pixels is equal to or below a given radius. We expect that each type of dynamic textures (e.g. corrosion evolution of different nanostructures) provides complex networks with specific topological features. In order to measure the topological features, we propose the spatial and temporal degrees. Thus, the proposed method is able to describe both appearance and motion of dynamic textures using the degrees. The spatial degree extracts appearance features while and motion features are extracted using the temporal degree. By using both measurements, the proposed method is able to discriminate dynamic textures with different appearances (spatial variance) and dynamic textures with different dynamics (temporal variance). Experimental results have shown that these measurements are discriminative for synthetic and real dynamic textures. In the traffic video classification, the proposed method achieved the highest correct classification rate compared to state-of-the-art.

The rest of the paper is organized as follows. Section 2 presents a brief review of the complex network theory. In Section 3, the proposed method for dynamic texture recognition is described in detail. The experiments conducted and the results for synthetic and real dynamic textures are presented in Section 4. Finally, conclusions and future works are given in Section 5.

2. Complex network theory

Complex network has recently attracted a lot of attention due to its multidisciplinary nature, taking place in areas such as biology [18], sociology [19], computer vision [20,21], and physics [22]. We can cite three main developments that have contributed for the complex network research [23]: (i) investigation of the small-world networks [24], which are characterized by small average node-to-node distance; (ii) investigation of the scale-free networks [25], which are networks whose degree distribution follows a power law; and (iii) identification of community structures in networks [26].

Conceptually, the complex network research lies at the intersection between graph theory and statistical mechanics [23]. The graph theory is a well-established area in computer science and mathematics, while the complex network theory has emerged from physics and sociology. Most research in complex networks involves two main steps: representation of the problem as a complex network followed by the analysis of its topological features obtained by a set of measurements. Given the measurements, it is possible to identify different categories of structures.

2.1. Complex network representation and measurements

The complex networks are represented by graphs. An undirected weighted graph $C=(N,E)$ is defined by a set of nodes $N=\{n_i\}$ and a set of edges $E=\{e_{n_i,n_j}\}$, where $e_{n_i,n_j} \in \mathbb{R}$ represents the weight of the connection between nodes n_i and n_j . In undirected graphs, the edge e_{n_i,n_j} is equal to edge e_{n_j,n_i} .

The most traditional measurements from complex networks are the node degree and node strength. The reader may consult [23] for a review of complex network measurements. The node degree $k(n_i)$ of a node n_i is given by the number of edges containing n_i , according to

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

The degree strength $kw(n_i)$ is given by the sum of weights of the edges containing n_i :

$$kw(n_i) = \sum_{n_j} \begin{cases} e_{n_i,n_j}, & e_{n_i,n_j} \in E \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

3. Complex network for dynamic textures

To describe the proposed method, let us consider a video that consists of a pair (\mathcal{V}, V) where \mathcal{V} is a set of pixels and V is a mapping

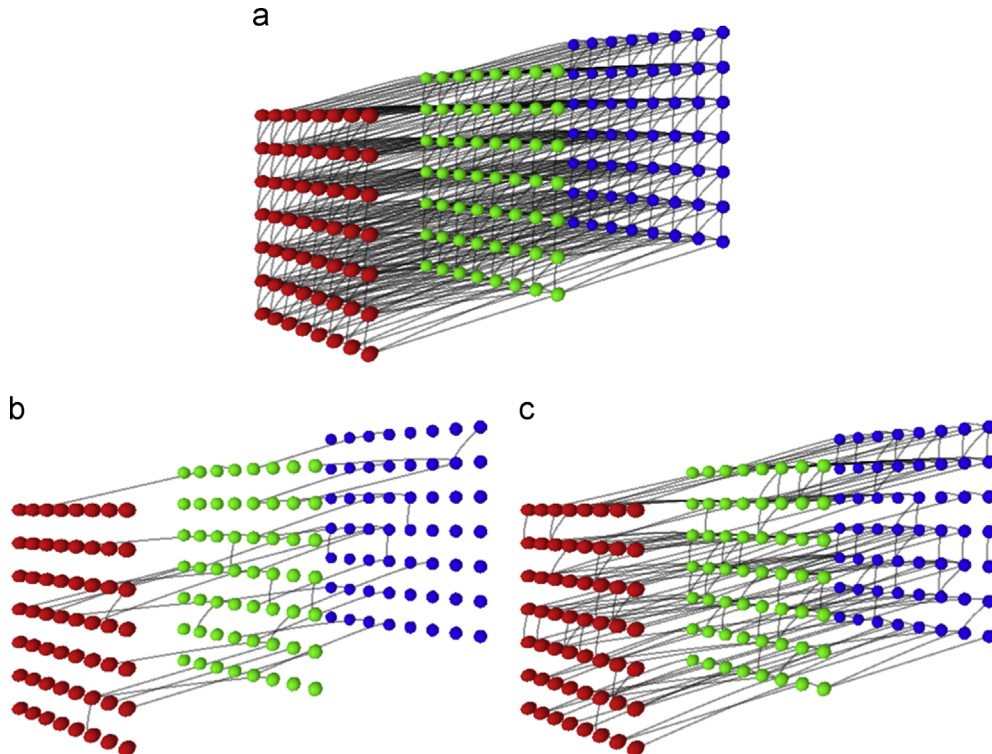


Fig. 1. Examples of function ϕ applied over a video composed by three frames. Figures (b) and (c) correspond to a transformation using thresholds $\tau_0 < \tau_1$. (a) Regular complex network. (b) τ_0 . (c) τ_1 .

that assigns to each pixel $i = (x_i, y_i, t_i) \in \mathcal{V}$ an intensity $V(i) \in [0, 255]$. To model the video as a complex network $C = (N, E)$, each pixel is mapped into a node of the set of nodes N . A regular complex network C^r is obtained by connecting two nodes n_i and n_j if the Euclidean distance between their related pixels is equal to or below a given radius r . For each edge $e_{n_i, n_j} \in E$, a weight is defined by the pixel intensity difference:

$$e_{n_i, n_j} = \begin{cases} |V(i) - V(j)| & \text{if } \text{dist}(i, j) \leq r \\ \text{NaN} & \text{otherwise} \end{cases} \quad (3)$$

where $\text{dist}(i, j) = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2 + (t_i - t_j)^2}$ is the Euclidean distance between pixels i and j . The pixel intensity difference $|V(i) - V(j)|$ is not affected by changes in mean luminance [27]. Thus, the weights of the complex network are invariant against gray-level shifts. In Fig. 1(a), we show an example of a regular complex network. In this example, the video has three frames and each node is connected with its neighbors in a given radius. All nodes have the same number of connections except those on the borders.

Given the regular complex network C^r , a function is applied over the edges to obtain a complex network that reveals different texture properties. The function $C_\tau^r = \phi(C^r, \tau)$ consists of removing edges according to their weights [27,28]. Edges with weight equals to or below τ are selected (Eq. (4)). Examples of this function using two thresholds $\tau_0 < \tau_1$ over the regular complex network are illustrated in Fig. 1. We can see that the number of connections increases as the threshold increases.

$$e_{n_i, n_j} = \begin{cases} e_{n_i, n_j} & \text{if } 0 \leq e_{n_i, n_j} \leq \tau \\ \text{NaN} & \text{otherwise} \end{cases} \quad (4)$$

The function $\phi(C^r, \tau)$ can be considered as a multiscale network analysis [27]. For each threshold τ , the regular complex network is transformed into a τ -scaled network. Compared to the regular complex network, a τ -scaled network presents different properties and reveals the structure and topology related to its scale. Small values of τ provide complex networks that models the details of the image represented by small sets of pixels (e.g. Fig. 1(b)). As the value of τ increases, the complex network models global information of the image, such as edges of the image. Thus, the proposed method aims at combining measures (e.g., average degree) from some τ -scaled networks to achieve a multiscale network analysis.

3.1. Spatial and temporal degrees

To measure the topological features from the transformed complex network C_τ^r , we propose the spatial and temporal degrees. The spatial degree $k_s(n_i)$ of node n_i is given by the number of edges that connects the node n_i to other nodes in the same

frame, according to

$$k_s(n_i) = \sum_{n_j} \begin{cases} 1, & e_{n_i, n_j} \in E \text{ and } t_i = t_j \\ 0 & \text{otherwise} \end{cases} \quad (5)$$

Similarly, the spatial node strength is given by

$$kw_s(n_i) = \sum_{n_j} \begin{cases} e_{n_i, n_j}, & e_{n_i, n_j} \in E \text{ and } t_i = t_j \\ 0 & \text{otherwise} \end{cases} \quad (6)$$

On the other hand, the temporal degree $k_t(n_i)$ of node n_i is the number of edges that connects n_i to other nodes in different frames, according to

$$k_t(n_i) = \sum_{n_j} \begin{cases} 1, & e_{n_i, n_j} \in E \text{ and } t_i \neq t_j \\ 0 & \text{otherwise} \end{cases} \quad (7)$$

The temporal node strength is given as follows:

$$kw_t(n_i) = \sum_{n_j} \begin{cases} e_{n_i, n_j}, & e_{n_i, n_j} \in E \text{ and } t_i \neq t_j \\ 0 & \text{otherwise} \end{cases} \quad (8)$$

Fig. 2(a) shows a complex network containing only the spatial degree k_s while Fig. 2(b) presents a network containing only the temporal degree k_t . The main idea of the spatial degree is that it can describe the appearance of the video, for example, different flags waving in the same dynamic. On the other hand, the temporal degree aims at describing different dynamics, such as the same flag waving in different ways. Thus, both degrees provide interesting features for classifying dynamic textures.

3.2. Feature vector

To describe the topology of the complex network, we present a feature vector composed of spatial and temporal degrees. This feature vector is calculated for a complex network C_τ^r built using radius r and threshold τ , according to

$$\begin{aligned} \varphi_\tau^r &= [\vartheta_\tau^r, \omega_\tau^r] \\ \vartheta_\tau^r &= [\bar{k}_s, \hat{k}_s, \tilde{k}_s, \bar{k}_t, \hat{k}_t, \tilde{k}_t] \\ \omega_\tau^r &= [kw_s, kw_t, \overline{kw_s}, \overline{kw_t}, k\overline{w_s}, k\overline{w_t}] \end{aligned} \quad (9)$$

where ϑ_τ^r is the statistics of the degree, ω_τ^r is the statistics of the node strength, $\bar{k}_s, \hat{k}_s, \tilde{k}_s$ are respectively the average, energy and entropy of the spatial degree (the k_t statistics are analogous). These statistics can be computed as follows:

$$\bar{k}_s = \frac{1}{|N|} \sum_{n_i} k_s(n_i), \quad \hat{k}_s = \sum_{n_i} k_s(n_i)^2, \quad \tilde{k}_s = - \sum_{n_i} k_s(n_i) \log k_s(n_i) \quad (10)$$

To obtain a multiscale network analysis, we concatenate feature vectors for different values of threshold τ . The values of the threshold are between the interval $\tau_0 \leq \tau \leq \tau_f$ and incremented by a constant τ_i . Thus, a combined feature vector φ^r , which

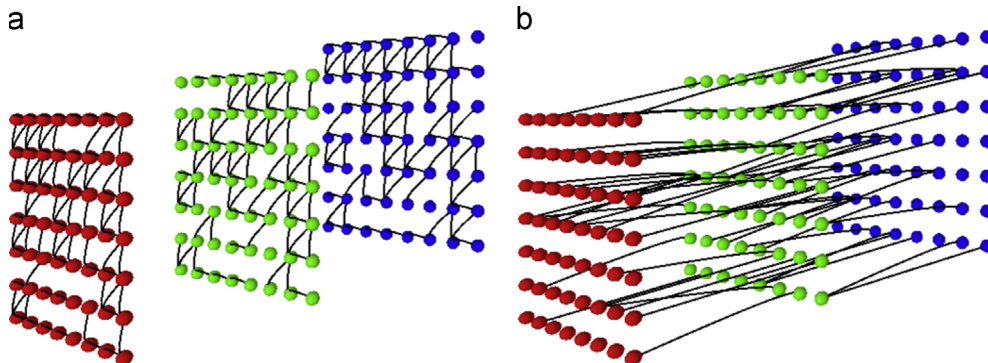


Fig. 2. Example of the spatial and temporal degrees. The spatial degree connects nodes in the same frame while the temporal degree connects nodes from different frames. (a) k_s , (b) k_t .

considers information from different transformed complex networks using radius r is given by

$$\varphi^r = [\varphi_{\tau_0}^r, \varphi_{\tau_0+\tau_i}^r, \dots, \varphi_{\tau_f}^r] \quad (11)$$

Finally, the feature vector φ is given by concatenating feature vectors φ^r for different values of r (Eq. (12)). The final vector contains information from networks transformed by different values of radius r and thresholds τ .

$$\varphi = [\varphi^{r_0}, \dots, \varphi^{r_n}] \quad (12)$$

4. Experiments and results

In this section, we present the experiments and the results obtained by the proposed method. Results for synthetic dynamic textures and a real application of monitoring traffic are shown in the following sections. We also present the influence of the method parameters and a comparison with the state-of-the-art in the applications.

Experiments have been conducted on three datasets:

- UCLA dataset [29,30]: this dataset consists of 200 videos divided into 50 dynamic textures. Examples of the first frame of eight dynamic textures can be seen in Fig. 3.
- Dyntex++ dataset [31]: the dataset used in this work consists of 1080 videos divided into 36 classes. Examples of the first frame of eight classes can be seen in Fig. 4.
- Traffic dataset [15]: this dataset consists of 254 videos with great variety of traffic patterns and weather conditions. The videos are divided into three classes – light, medium, and heavy traffic. Examples of the three classes can be seen in Fig. 5.

The feature vector proposed in this work is extracted for each video of the datasets and used in a classification system to evaluate the methods. The classification was performed using the Nearest-Neighbor classifier in the 10-fold cross-validation scheme. This classifier was chosen due its simplicity, highlighting the importance of the features.

4.1. Synthetic dynamic textures

In Fig. 6, we have plotted the average spatial degree k_s and the average temporal degree k_t versus thresholds τ for synthetic dynamic textures obtained from [32]. Fig. 6(a) shows the synthetic dynamic textures used in the plots, where the squares indicate the pixels used to calculate the degree measurements. The samples differ only in the appearance, sharing the same dynamics. As we can see from Fig. 6(b), the average spatial degree presented two distinct curves, which corroborates that the spatial degree explores spatial cues. On the other hand, the plot in Fig. 6(c) shows that the average temporal degree did not provide distinguishable curves, since the dynamic remains the same all over the frames.

To show the discrimination power of the temporal degree, Fig. 7(a) shows two synthetic dynamic textures that are identical in appearance, but differ in the dynamics. Both synthetic dynamic textures are indicated by the squares. We can note that, even for human beings, this particular identification task is quite difficult. For these dynamic textures, average spatial degree k_s can not discriminate both dynamic textures, since the appearance is the same (Fig. 7(b)). However using the average temporal degree k_t (Fig. 7(c)), the synthetic dynamic textures can be discriminated thanks to the temporal properties of the dynamic textures. Using these synthetic dynamic textures, we corroborate the importance of using both measures in the dynamic texture representation.

4.2. Analysis of parameters

We first analyze the influence of each parameter of the proposed method on the datasets. The four parameters are the radius r , the initial threshold τ_0 , the incremental threshold τ_i and the final threshold τ_f . In the experiments below, we present the results for configurations of these parameters.

The results for different values of initial threshold τ_0 , responsible for limiting the scale of the method, are presented in Fig. 8(a). As we can see, the proposed method achieved the highest classification rate for $\tau_0 = 8, 8, 1$ for Dyntex++, traffic and UCLA datasets, respectively. Low values of τ_0 indicate that the details of fine-scales are important in the recognition stage.

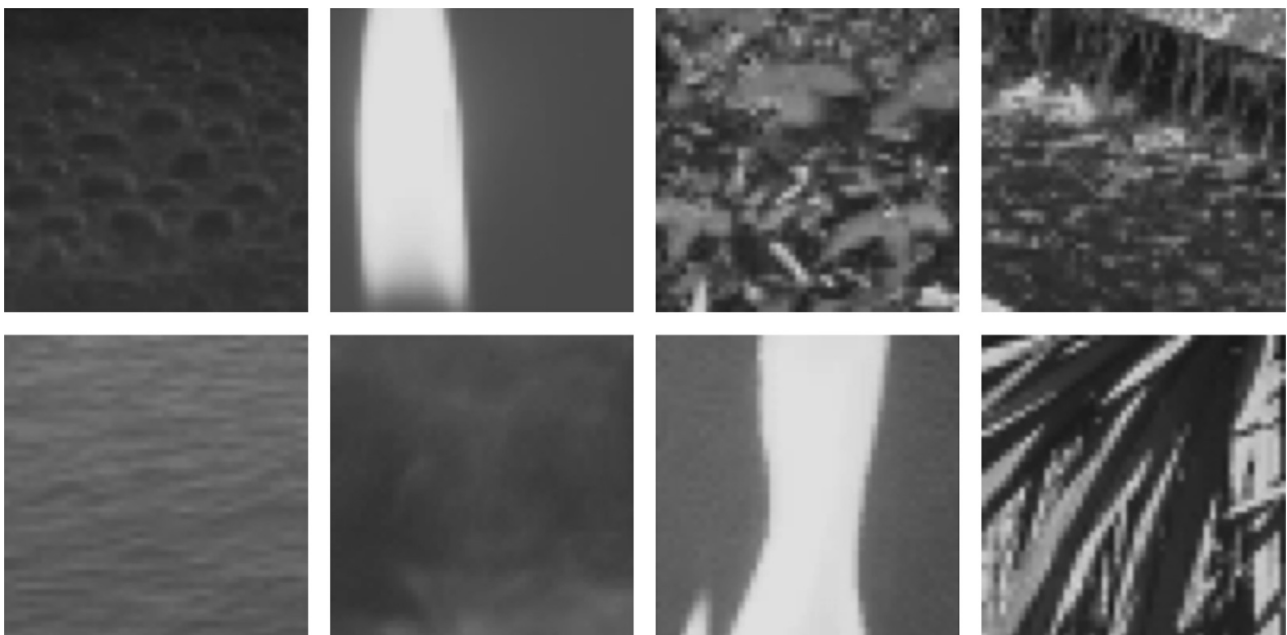


Fig. 3. Example of the first frame of eight classes from UCLA dataset. The classes are, respectively, boiling, candle, flowers, fountain, sea, smoke, fire and plant.



Fig. 4. Examples of the first frame of eight classes from Dyntex++ dataset.



Fig. 5. Classes of the traffic dataset. The 254 videos are divided into 3 three classes light, medium and heavy traffic shown in the first, second and third columns respectively.

Fig. 8(b) shows a plot for evaluating the incremental threshold τ_i . The highest classification rate for the traffic dataset is obtained using $\tau_i = 11$. On the other hand, high values for τ_i are required on Dyntex++ and UCLA datasets, respectively $\tau_i = 33$ and 43. As

expected, high values of τ_i provided good classification rates, since it allows the proposed method to explore a better range of scales. On the Dyntex++ and UCLA datasets, the range of thresholds is greater to characterize the variety of dynamic textures.

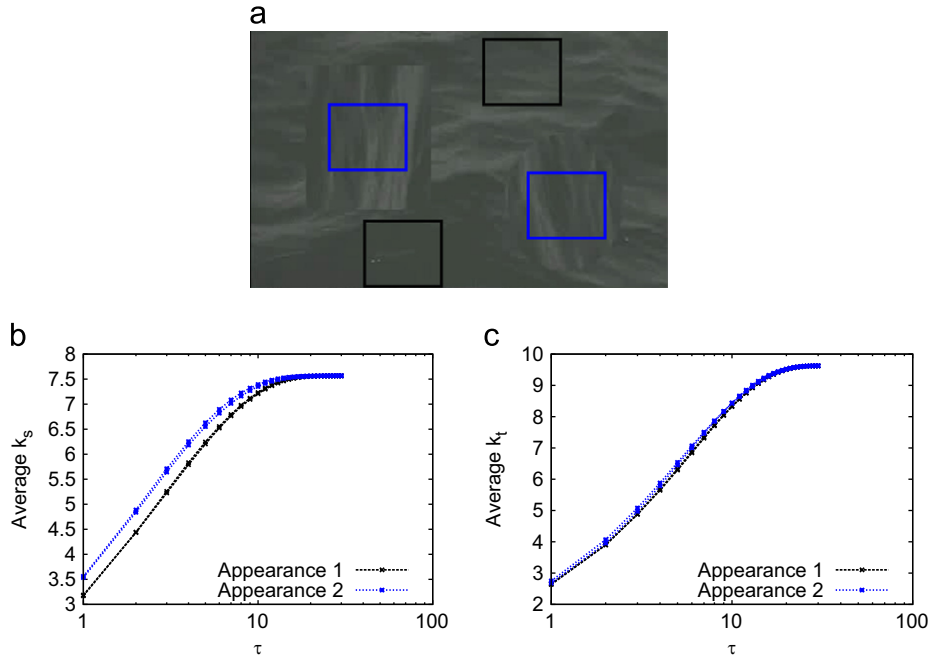


Fig. 6. Threshold τ versus average k_s and average k_t for synthetic dynamic textures with different appearances. (a) Dynamic textures with different appearances. (b) Average spatial degree. (c) Average temporal degree.

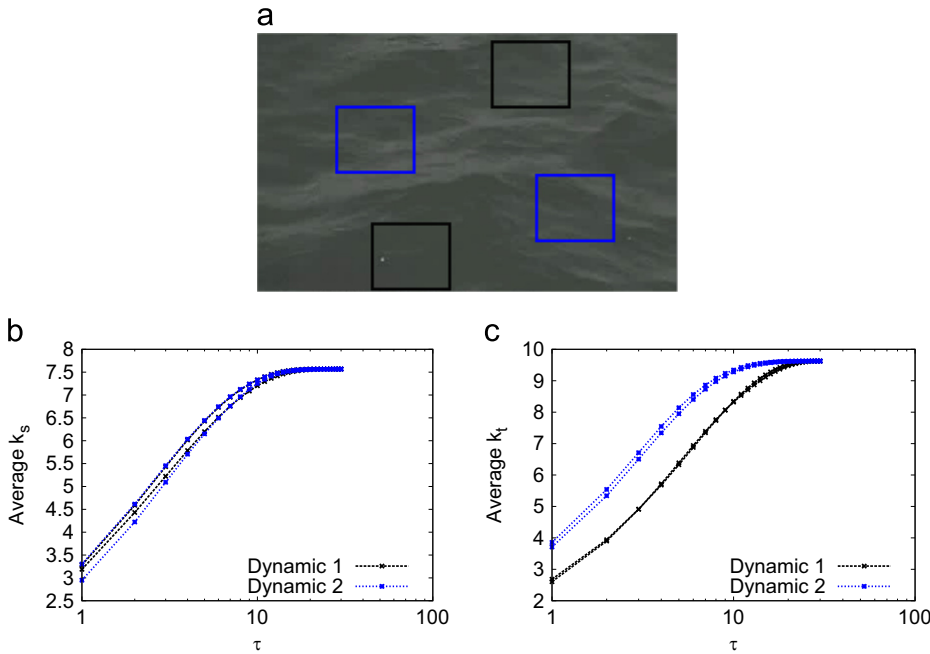


Fig. 7. Threshold τ versus average k_s and average k_t for synthetic dynamic textures with different dynamics. (a) Dynamic textures with different dynamics. (b) Average spatial degree. (c) Average temporal degree.

The final threshold τ_f is evaluated in Fig. 8(c) as a function of the number of thresholds n_τ , $\tau_f = \tau_0 + n_\tau * \tau_i$. The maximum correct classification rate is achieved for $n=3, 4, 3$ on traffic, Dyntex++ and UCLA datasets, respectively. We can observe that few scales already provided high classification rates, an important result with respect to the computational time. Using these results, we can determine the suitable set of thresholds for each dataset.

Finally, the radius r determines the zone of connection between nodes. The evaluation of this parameter is shown in Fig. 8(d). Note that the best result on the traffic dataset is obtained for the concatenation of a few radiuses, $r = \{\sqrt{1}, \sqrt{2}, \sqrt{4}\}$. For the Dyntex++ and UCLA, the set of radiuses that provided the best result is $r = \{\sqrt{1}, \sqrt{2}, \sqrt{4}, \sqrt{5}, \sqrt{8}, \sqrt{9}, \sqrt{10}\}$.

Given the above experiments and results, we can conclude that a greater range of thresholds and radiuses are needed to cover the wide variety of dynamic textures from the Dyntex++ and UCLA datasets. Since the traffic dataset is composed of similar classes (highway traffic), a smaller range already provided good results.

4.3. Comparison with other methods

To show the effectiveness of the proposed method against the state-of-the-art, Tables 1–3 provide a comparison of existing methods in the Dyntex++, UCLA and traffic datasets, respectively. The acronyms NF, CCR and AUC stand for Number of Features, Correct Classification Rate and Area Under ROC Curve (ROC – receiver operating characteristic).

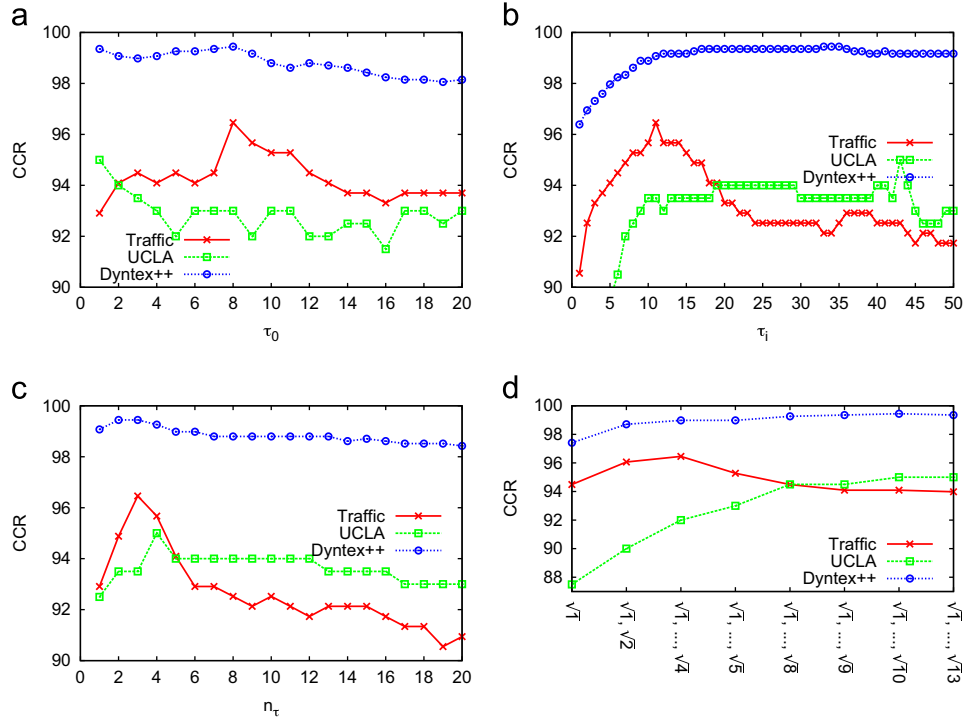


Fig. 8. Analysis of parameters in the traffic, Dyntex++ and UCLA datasets. (a) τ_0 . (b) τ_1 . (c) n_t . (d) r .

Table 1

Classification results for the UCLA dataset. NF, CCR and AUC stand for number of features, correct classification rate and area under ROC curve (ROC – receiver operating characteristic), respectively.

Method	NF	CCR	Precision	Recall	AUC
RI-VLBP	16,384	77.50(± 8.98)	0.78(± 0.10)	0.78(± 0.09)	0.89(± 0.05)
LBP-TOP	768	95.00(± 4.44)	0.96(± 0.05)	0.95(± 0.04)	0.97(± 0.02)
DPSW	75	95.00(± 4.78)	0.96(± 0.05)	0.95(± 0.05)	0.97(± 0.02)
Proposed method	420	95.00(± 5.19)	0.96(± 0.06)	0.95(± 0.05)	0.97(± 0.03)

Table 2

Classification results for the Dyntex++ dataset. NF, CCR and AUC stand for number of features, correct classification rate and area under ROC curve (ROC – receiver operating characteristic), respectively.

Method	NF	CCR	Precision	Recall	AUC
RI-VLBP	16,384	97.87(± 1.47)	0.98(± 0.01)	0.98(± 0.01)	0.99(± 0.01)
LBP-TOP	768	99.35(± 0.72)	0.99(± 0.01)	0.99(± 0.01)	1.0(± 0.00)
DPSW	75	98.33(± 1.26)	0.98(± 0.01)	0.98(± 0.01)	0.99(± 0.01)
Proposed method	336	99.44(± 0.67)	0.99(± 0.01)	0.99(± 0.01)	1.0(± 0.00)

Table 3

Classification results for the traffic dataset. NF, CCR and AUC stand for number of features, correct classification rate and area under ROC curve (ROC – receiver operating characteristic), respectively.

Method	NF	CCR	Precision	Recall	AUC
RI-VLBP	16,384	93.31(± 4.34)	0.93(± 0.05)	0.93(± 0.04)	0.96(± 0.04)
LBP-TOP	768	93.70(± 4.70)	0.94(± 0.05)	0.94(± 0.05)	0.95(± 0.04)
DPSW	75	93.70(± 4.83)	0.94(± 0.03)	0.94(± 0.03)	0.96(± 0.03)
Proposed method	144	96.46(± 4.10)	0.97(± 0.04)	0.97(± 0.04)	0.98(± 0.03)

Besides these three performance measures, the table also presents the Precision and Recall, which are commonly used in the performance evaluation of algorithms. We can note that the proposed method provided similar or better results than the state-of-the-art methods: Rotation Invariant Volumetric Local Binary Patterns (RI-VLBP) [33], Deterministic Partially Self-avoiding Walks (DPSW) [9], and Local Binary Patterns – Three Orthogonal Planes (LBP-TOP) [34]. We have used the

default parameters for each state-of-the-art methods as described in the original papers. The number of features for RI-VLBP and LBP-TOP can be calculated by Equations 2^{3P+2} and 3×2^P [36], where P is the number of neighbors. For $P=4$ and $P=8$ (default parameters), it is obtained 16,384 and 768 features for RI-VLBP and LBP-TOP, respectively.

In Fig. 9, we present the best previous correct classification rates reported on the traffic dataset and their respective confusion matrices.

Besides the LBP-TOP and the proposed method, the methods proposed by Derpanis and Wildes [35] and by Chan and Vasconcelos [15] are taken in the comparisons. It is worth mentioning that the methods in [35,15] have used different experimental protocols and consequently the comparison should be taken cautiously. For example, the work of Chan and Vasconcelos has used the support vector machine classifier (SVM), which is known to provide superior results than the Nearest

Neighbor classifier (NN) for the same set of features [36,37]. Since our focus is on representation, we have chosen a simple classifier (Nearest Neighbor) to evidence the importance of the features in the classification. Even using the NN classifier, the proposed method achieved superior results compared to the best previous correct classification rates. The proposed method improves correct classification rate from 95.28% obtained by Derpanis to 96.46%. Using SVM classifier, our

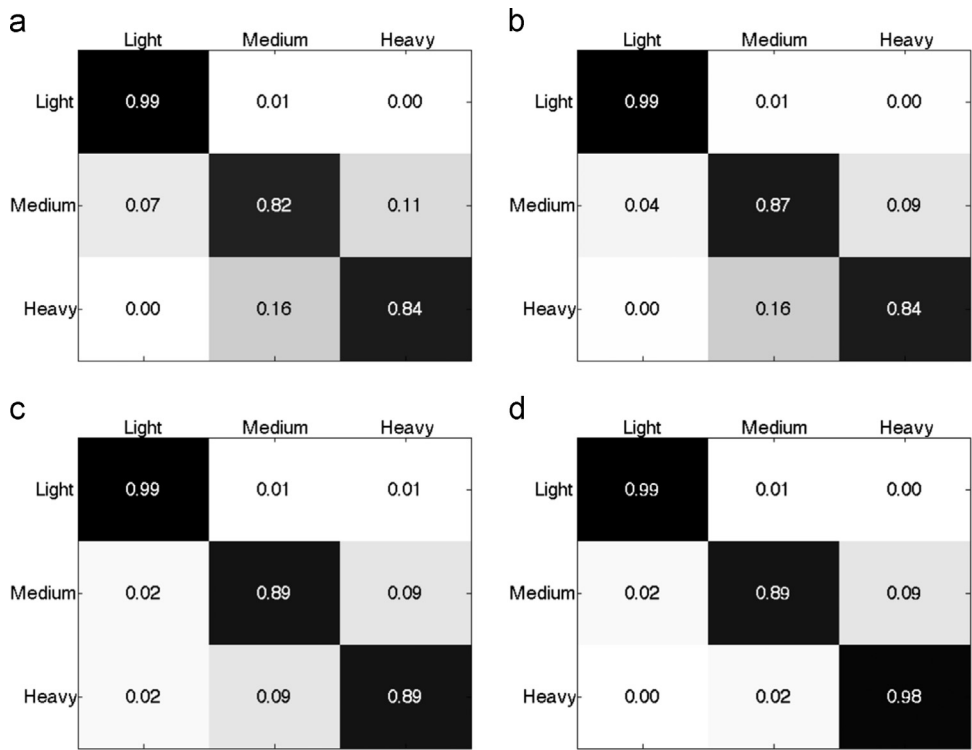


Fig. 9. Confusion matrices for methods reported in the literature on the traffic dataset. (a) LBP-TOP, 93.70%. (b) Chan, 94.50%. (c) Derpanis, 95.28%. (d) Proposed method using SVM classifier, 97.24%.



Fig. 10. Example of the first frame of three videos from the medium traffic that were incorrectly classified as heavy traffic.

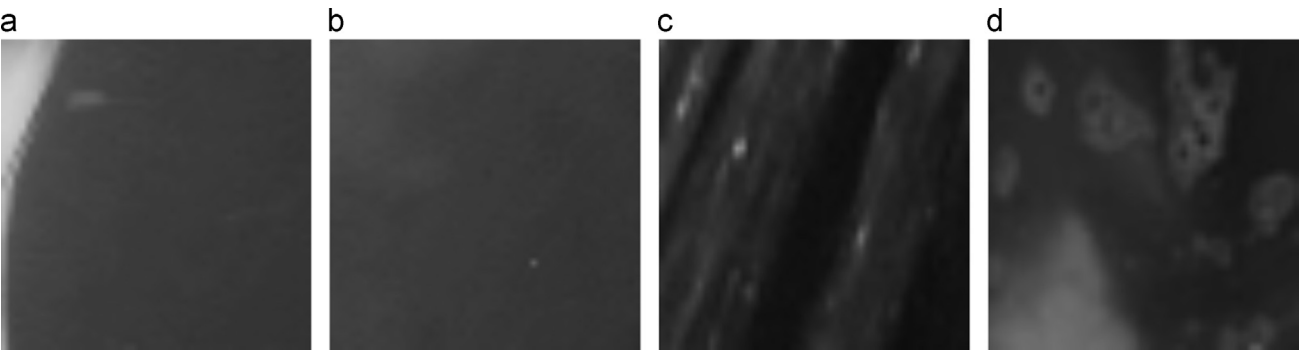


Fig. 11. Example of the first frame of four videos from Dyntex++ and UCLA datasets incorrectly classified. (a) Fire. (b) Smoke. (c) Shower. (d) Foam.

method improves its correct classification rate from 96.46 (± 4.10) to 97.24 (± 3.62).

As can be seen from the confusion matrices, the medium traffic class is the most difficult to describe and classify. Fig. 10 shows the first frame of three videos from medium traffic that were incorrectly classified as heavy traffic by the proposed method. The appearance of these videos is similar to the heavy traffic, although the highway speed (temporal information) is similar to the videos from the medium traffic. Since our method extracts both features from the videos through spatial and temporal degrees, these three examples could be classified as either of two traffic conditions.

Fig. 11 shows the first frame of videos from Dyntex++ and UCLA datasets incorrectly classified. For example, the video of Fig. 11(a) was incorrectly classified as class Smoke (Fig. 11(b)). Similarly, the video of Fig. 11(c) was incorrectly classified as class Foam (Fig. 11(d)). As we can see, most of pixels present similar behavior in Fig. 11(a) and (b). As the method calculates the spatial and temporal average degrees, a large amount of similar pixels dominate the average.

4.4. Computational complexity

The proposed method represents a dynamic texture as a complex network, where each pixel is mapped into a node. Considering a video with $N \times N \times T$ pixels, each node is connected to all neighbors inside a $(2r+1) \times (2r+1) \times (2r+1)$ window. Therefore $O((2r+1) \times (2r+1) \times (2r+1) \times N^2 \times T)$ operations are needed to build the complex network. In video applications, the radius is much smaller than the number of pixels, i.e., $r \ll N^2 \times T$. Thus, we can ignore r in the complexity, leading to a computational complexity of $O(N^2 \times T)$. To build a complex network with $48 \times 48 \times 75$ nodes and radius $r = \sqrt{5}$, it took on average 0.053 s using 1.3 GHz Intel (R) Core i5, 4 GB RAM and 64-bit Operating System. These results indicate competitive running time for real time applications.

5. Conclusion

Complex network is a powerful and flexible tool to model real-world structures and characterize their topology. In this paper, we propose a novel method for dynamic texture recognition based on complex networks. After the complex network representation, spatial and temporal measurements are extracted from the complex network. These measurements proved to be powerful to characterize the appearance and dynamics of textures. Experiments in synthetic and real dynamic textures have demonstrated the effectiveness of the proposed method. In these experiments, the proposed method provided better results than state-of-the-art in the traffic dynamic textures.

Acknowledgments

W.N.G. acknowledges support from FUNDECT. B.B.M. acknowledges support from FAPESP under grant number 2011/02918-0. O. M.B. gratefully acknowledges the financial support of CNPq (National Council for Scientific and Technological Development, Brazil) (Grant nos. 308449/2010-0 and 473893/2010-0) and FAPESP (Grant no. 2011/01523-1).

References

- [1] W. Hu, N. Xie, L. Li, X. Zeng, S.J. Maybank, A survey on visual content-based video indexing and retrieval, *IEEE Trans. Syst. Man Cybern. C* 41 (6) (2011) 797–819.
- [2] S. Fazekas, D. Chetverikov, Analysis and performance evaluation of optical flow features for dynamic texture recognition, *SP: IC 22 (7–8)* (2007) 680–691.
- [3] R. Polana, R.C. Nelson, Temporal texture and activity recognition, in: *Motion-Based Recognition*, 1997, (Chapter 5).
- [4] R. Fablet, P. Bouthemy, Motion recognition using nonparametric image motion models estimated from temporal and multiscale cooccurrence statistics, *IEEE Trans. Pattern Anal. Mach. Intell.* 25 (12) (2003) 1619–1624.
- [5] S. Dubois, R. Péteri, M. Ménard, A comparison of wavelet based spatio-temporal decomposition methods for dynamic texture recognition, in: *Fourth Iberian Conference on Pattern Recognition and Image Analysis*, Springer-Verlag, Berlin, Heidelberg, 2009, pp. 314–321.
- [6] H. Zhong, J. Shi, M. Visontai, Detecting unusual activity in video, in: *IEEE Conference on Computer Vision and Pattern Recognition*, 2004, pp. 819–826.
- [7] P. Dollar, V. Rabaud, G. Cottrell, S. Belongie, Behavior recognition via sparse spatio-temporal features, in: *ICCCN '05: Proceedings of the Fourteenth International Conference on Computer Communications and Networks*, IEEE Computer Society, Washington, DC, USA, 2005, pp. 65–72.
- [8] A.B. Chan, N. Vasconcelos, Layered dynamic textures, *IEEE Trans. Pattern Anal. Mach. Intell.* 31 (10) (2009) 1862–1879.
- [9] W.N. Gonçalves, O.M. Bruno, Dynamic texture analysis and segmentation using deterministic partially self-avoiding walks, *Expert Syst. Appl.* 40 (11) (2013) 4283–4300 (<http://dx.doi.org/10.1016/j.eswa.2012.12.092>).
- [10] G. Doretto, A. Chiuso, Y.N. Wu, S. Soatto, Dynamic textures, *Int. J. Comput. Vis.* 51 (2) (2003) 91–109.
- [11] A.B. Chan, N. Vasconcelos, Classifying video with kernel dynamic textures, in: *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2007, pp. 1–6.
- [12] A.B. Chan, N. Vasconcelos, Modeling, clustering, and segmenting video with mixtures of dynamic textures, *IEEE Trans. Pattern Anal. Mach. Intell.* 30 (5) (2008) 909–926. <http://dx.doi.org/10.1109/TPAMI.2007.70738>.
- [13] M. Szummer, R.W. Picard, Temporal texture modeling, in: *ICIP*, 1996, pp. III: 823–826.
- [14] R. Costantini, L. Sbaiz, S. Susstrunk, Higher order SVD analysis for dynamic texture synthesis, *IEEE Trans. Image Process.* 17 (1) (2008) 42–52. <http://dx.doi.org/10.1109/TIP.2007.910956>.
- [15] A.B. Chan, N. Vasconcelos, Classification and retrieval of traffic video using auto-regressive stochastic processes, in: *IEEE Intelligent Vehicles Symposium*, 2005, pp. 771–776.
- [16] M. Fujii, T. Horikoshi, K. Otsuka, S. Suzuki, Feature extraction of temporal texture based on spatiotemporal motion trajectory, in: *ICPR*, vol. II, 1998, pp. 1047–1051.
- [17] D. Chetverikov, R. Péteri, A brief survey of dynamic texture description and recognition, in: *International Conference on Computer Recognition Systems, Advances in Soft Computing*, vol. 30, Springer, 2005, pp. 17–26.
- [18] A.-L. Barabási, Z.N. Oltvai, Network biology: understanding the cell's functional organization, *Nat. Rev. Genet.* 5 (2) (2004) 101–113. <http://dx.doi.org/10.1038/nrg1272>.
- [19] M.E.J. Newman, J. Park, Why social networks are different from other types of networks, *Phys. Rev. E (Stat. Nonlinear Soft Matter Phys.)* 68 (3) (2003) 036122.
- [20] A.R. Backes, D. Casanova, O.M. Bruno, A complex network-based approach for boundary shape analysis, *Pattern Recognit.* 42 (1) (2009) 54–67. <http://dx.doi.org/10.1016/j.patcog.2008.07.006>.
- [21] W. Gonçalves, A. Martinez, O. Bruno, Complex network classification using partially self-avoiding deterministic walks, *Chaos* 22 (3) (2012) 033139.
- [22] L.A.N. Amaral, J.M. Ottino, Complex networks: augmenting the framework for the study of complex systems, *Eur. Phys. J. B* 38 (2) (2004) 147–162.
- [23] L.F. Costa, F.A. Rodrigues, G. Traverso, P.R.V. Boas, Characterization of complex networks: a survey of measurements, *Adv. Phys.* 56 (1) (2006) 167–242.
- [24] D.J. Watts, S.H. Strogatz, Collective dynamics of small-world networks, *Nature* 393 (6684) (1998) 440–442. <http://dx.doi.org/10.1038/30918>.
- [25] A.-L. Barabási, R. Albert, Emergence of scaling in random networks, *Science* 286 (5439) (1999) 509–512. <http://dx.doi.org/10.1126/science.286.5439.509>.
- [26] M. Girvan, M.E.J. Newman, Community structure in social and biological networks, *Proc. Natl. Acad. Sci.* 99 (12) (2002) 7821–7826. <http://dx.doi.org/10.1073/pnas.122653799>.
- [27] W.N. Gonçalves, A.R. Backes, A.S. Martinez, O.M. Bruno, Texture descriptor based on partially self-avoiding deterministic walker on networks, *Expert Syst. Appl.* 39 (15) (2012) 11818–11829. <http://dx.doi.org/10.1016/j.eswa.2012.01.094>.
- [28] W.N. Gonçalves, J. de Andrade Silva, O.M. Bruno, A rotation invariant face recognition method based on complex network, in: *Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications, Lecture Notes in Computer Science*, vol. 6419, Springer, Berlin/Heidelberg, 2010, pp. 426–433.
- [29] A.B. Chan, N. Vasconcelos, Probabilistic kernels for the classification of auto-regressive visual processes, in: *Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)* – vol. 1, CVPR'05, IEEE Computer Society, Washington, DC, USA, 2005, pp. 846–851. <http://dx.doi.org/10.1109/CVPR.2005.279>.
- [30] P. Saisan, G. Doretto, Y. N. Wu, S. Soatto, Dynamic texture recognition, in: *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2001*, vol. 2, IEEE, 2001, pp. II–58.
- [31] B. Ghanem, N. Ahuja, Maximum margin distance learning for dynamic texture recognition, in: *Proceedings of the Eleventh European Conference on Computer Vision: Part II, ECCV'10*, Springer-Verlag, Berlin, Heidelberg, 2010, pp. 223–236.
- [32] G. Doretto, D. Cremers, P. Favaro, S. Soatto, Dynamic texture segmentation, in: *Proceedings of the International Conference on Computer Vision*, vol. 2, 2003, pp. 1236–1242.

- [33] G. Zhao, M. Pietikäinen, Improving rotation invariance of the volume local binary pattern operator, in: MVA, 2007, pp. 327–330.
- [34] G. Zhao, M. Pietikäinen, Dynamic texture recognition using local binary patterns with an application to facial expressions, *IEEE Trans. Pattern Anal. Mach. Intell.* 29 (6) (2007) 915–928.
- [35] K.G. Derpanis, R.P. Wildes, Classification of traffic video based on a spatio-temporal orientation analysis, in: Proceedings of the 2011 IEEE Workshop on Applications of Computer Vision (WACV), WACV '11, IEEE Computer Society, Washington, DC, USA, 2011, pp. 606–613. <http://dx.doi.org/10.1109/WACV.2011.5711560>.
- [36] J. Zhang, M. Marszałek, S. Lazebnik, C. Schmid, Local features and kernels for classification of texture and object categories: a comprehensive study, *Int. J. Comput. Vision* 73 (2) (2007) 213–238. <http://dx.doi.org/10.1007/s11263-006-9794-4>.
- [37] B. Caputo, E. Hayman, P. Mallikarjuna, Class-specific material categorisation, in: Proceedings of the Tenth IEEE International Conference on Computer Vision – vol. 2, ICCV '05, IEEE Computer Society, Washington, DC, USA, 2005, pp. 1597–1604. <http://dx.doi.org/10.1109/ICCV.2005.54>.



Wesley Nunes Gonçalves received his B.Sc. in Computer Engineering in 2007, and his M.Sc. and Ph.D. degrees in Computer Science from the University of São Paulo in 2010 and 2013, respectively. He is currently an associate professor in the Federal University of Mato Grosso do Sul, Brazil. His current research interests include texture analysis, scene recognition, object tracking and pattern recognition.



Bruno Brandoli Machado received his B.Sc. in Computer Engineering, in 2007, and his M.Sc. degree from University of São Paulo, Brazil, in 2010. He is currently an associate professor in the Federal University of Mato Grosso do Sul and a Ph.D. candidate at Institute of Mathematics and Computer Science, at University of São Paulo, Brazil. His current research interests include texture analysis, complex systems, non-linear partial differential equations and pattern recognition.



Odemir Martinez Bruno is an associate professor at the Institute of Physics of S. Carlos at the University of S. Paulo in Brazil. He received his B.Sc. in Computer Science in 1992, from the Piracicaba Engineering College (Brazil), his M.Sc. in Applied Physics (1995) and his Ph.D. in Computational Physics (2000) at the University of S. Paulo (Brazil). His fields of interest include Computer Vision, Image Analysis, Computational Physics, Pattern Recognition and Bioinformatics. He is an author of many papers (journal and proceedings) and several book chapters, co-author of two books (*Optical and Physiology of Vision: a multidisciplinary approach* (Portuguese only) and *Internet programming with PHP* (Portuguese only)) and an inventor of five patents.