A Consensus Framework for Segmenting Video with Dynamic Textures

Lazhar Khelifi[1,2] and Max Mignotte[3]
[1]Ecole de Technologie Suprieure, University of Quebec, Canada
[2]INRS-EMT, Institut National de la Recherche Scientifique, Quebec, Canada
[3]Department of Computer Science and Operations Research, University of Montreal, Montreal, Canada
lazhar.khelifi@emt.inrs.ca, mignotte@iro.umontreal.ca

Abstract

Dynamic texture (DT) segmentation is the problem of clustering into groups various characteristics and phenomena that reproduce in both time and space, assigning a unique label to each group or region. Though this problem is highly complex, it has recently become the focus of considerable interest. This paper presents a simple and effective fusion framework for dynamic texture segmentation, whose objective is to combine multiple and weak region-based segmentation maps to get a final better segmentation result. The different label fields to be fused, are given by a simple clustering technique applied to an input video (based on three orthogonal planes xy, xt and yt). This is using as features a set of values of the requantized local binary patterns (LBP) histogram around the pixel to be classified. Promising preliminary experimental results have been achieved by our method on the challenging SynthDB dataset. Compared to existing dynamic texture segmentation approaches that require estimation of parameters or training classifiers, our method is easy to implement, simple and has few parameters.

1. Introduction

Combining texture and motion leads to a certain type of motion pattern known as dynamic textures (DT) or texture movies [1]. Compared to the static case, the segmentation of the dynamic texture is a highly challenging problem. This is because of the unknown spatial and temporal extend of dynamic scenes in the real world, which include, for example; smoke, sea waves, fire, foliage, etc [2].

In recent years, research on dynamic texture segmentation has become very popular. This research has produced interesting and various methods. Doretto *et al.* [3] proposed a technique to segment an image sequence into regions based on their spatio-temporal statistics. It works by modeling the spatio-temporal dynamics in each region with Gauss-Markov models, followed by a variational optimization framework to infer the model parameters as well as the boundary of the regions. However, this method has an assumption that the regions are constant in time and are slowly changed related to the irradiance within each region. Vidal *et al.*[4] addressed this problem based on an optical flow estimation, followed by a generalized principal component analysis (GPCA) step which segments a video by clustering pixels with similar trajectories in time. Nevertheless, in this model, a perceptual decomposition into more than two regions is not supported. Wattanachote *et al.* [5] proposed a new semiautomatic dynamic texture segmentation method based on the motion vectors derived from Farnebäck's [1] method [6]. A key limitation of this system is that required the intervention of the user to choose target objects and also to adjust the result in order to produce high-quality output. Nguyen *et al.* [7] presented a new unsupervised feature selection dynamic mixture model (FSDTM) for motion segmentation. The main advantage of their method is that does not require knowledge of any class labels. However, in this work the EM algorithm is required to maximize the data log-likelihood and also to optimize the parameters. In a major advance, Teney *et al.* [8] combined a filter-based motion features with a supervised learning approach. Recently, deep learning methods have been successfully applied to dynamic texture segmentation due to the immense effectiveness of convnets. In particular, a convolutional neural networks (CNNs) applied on three orthogonal planes xy, xt and yt, is proposed by Andrearczyk *et al.* [9]. The main weakness in their study is that the training of independent CNNs on three orthogonal planes, and the combining of their outputs makes the process more computationally complex.

Motivated by the aforementioned observations, we intro-

[1]An algorithm for estimating dense optical flow based on modeling the neighborhoods of each pixel by quadratic polynomials.

duce a new fusion model for dynamic texture segmentation. Our model aims to combine multiple and weak segmentation results to achieve a more reliable and final refined segmentation. These initial segmentation results are estimated from different slices (i.e., frames) Also, to overcome the drawbacks of previous techniques we propose a simple energy-minimization model. This energy function is originated from the global consistency error (GCE). The GCE criterion is a perceptual measure which takes into account the inherent multiscale nature of an image segmentation by measuring the level of refinement existing between two spatial partitions. In addition, to optimize our energy model, we propose a modified local optimization procedure derived from the iterative conditional modes (ICM) algorithm.

2. Proposed Method

The method described here is unsupervised, simple, and performed through five steps. In the initial stage of the process, a set of frames is generated by slicing the dynamic texture data (i.e. video). During the second step, a feature extraction process is realized and built for each frame. In the third step, a different dimensionality reduction based on different seeds, is applied over the extracted histogram related to each pixel. Then, a set of initial segmentations is generated by a clustering technique. As soon as these steps have been carried out, in the fourth step, a fusion scheme is done through the set of segmentations and iteratively optimized by a deterministic algorithm. The high-level overview of our method is shown in Fig. 1.

2.1. Slicing the Dynamic Texture Data

To take advantage of the complementarity of the three orthogonal planes on the input video sequence V, we perform a simple slicing task. Firstly, in the temporal xt plane, we generate h slices (i.e., frames), equally spaced on the y axis. In particular, a slice of the xt plane represents the evolution of a row of pixels over time along the video. Secondly, in the spatial xy plane, we simply generate w slices equally spaced in the temporal axis t from V. And thirdly, in the temporal yt plane, we generate m slices, equally spaced on the x axis. Explicitly, a slice of the yt plane represents the evolution of a column of pixels over time along the video sequence. Finally, after this slicing step, we obtain $h \times w \times m$ frames separated into three sets.

2.2. LBP Representation

To describe texture more effectively, we apply the local binary pattern (LBP) operator over each generated frame. The LBP operator aims to represent statistics of micro patterns contained in an image (i.e., frame in our case) by encoding the difference between the pixel value of the center point and those of its neighbors [10]. Denote as F a gray frame and let q_c be the value of the center pixel c of a local neighborhood. Let q_p $(p = 0, ..., P-1)$ be the values of P equally spaced pixels on a circle of radius R that form a circularly symmetric set of neighbors. If the coordinates of q_c are $(0, 0)$, then the coordinates of q_p are defined by $(R\sin(\frac{2\pi p}{P}), R\cos(\frac{2\pi p}{P}))$. In particular, a bilinear interpolation is used to estimate the values of neighbors which do not fall exactly in the center of a pixel. The LBP operator on this pixel (c) is then given by:

$$LBP_{P,R} = \sum_{p=0}^{P-1} s(q_s - q_c)2^p,\ s(x) = \begin{cases} 1 & , x \geq 0 \\ 0 & , x < 0 \end{cases} \quad (1)$$

2.3. Generation of the Segmentation Ensemble

Once the LBP representation step is performed, we project all pixels of each LBP-frame onto the xy plane. Then, at each frame, we compute around each pixel to be estimated a local requantized LBP histogram (on an overlapping squared fixed-size Nw = 7 neighborhood). Moreover, for each pixel $p^i_{(x,y)}$ we estimate a high-dimensional histogram by concatenating all local histograms related to the same pixel at each time t. In addition, we execute a dimensionality reduction algorithm on the high-dimensional histogram with different seeds. We adopt this choice to reduce the noise or irrelevant information of the data, and also to achieve more variability. Then, we pass the various low-dimensional histograms[2](related to the different seeds) to the clustering algorithm to generate groups. At this point, we resort to the useful k-means-based clustering technique [11]. We have adopted this choice to ensure a reduced computational time and cost for this important step.

2.4. Fusion Based on The Global Consistency Error Criterion

After the generation of the segmentation set has been done, we can then combine all these weak segmentations based on a new criterion called the global consistency error (GCE).

2.4.1 Global Consistency Error Criterion

This criterion is derived from the so-called local refinement error (LRE) which measures the degree of refinement between two segmentations [12]. In this sense, segmentations are considered to be consistent, since they could represent the same segmented image at different scales (or level of details) [13] [14]. Denote as n the number of pixels within the frame F and let $\Phi_\mu = \{s^1_\mu, s^2_\mu, \ldots, s^{nb_\mu}_\mu\}$ & $\Phi_\nu = \{s^1_\nu, s^2_\nu, \ldots, s^{nb_\nu}_\nu\}$ be, two segmentation results of the same frame to be compared, nb_μ being the number of

[2]The size of the final feature vector is 20 times smaller than the size of the original high-dimensional vector.

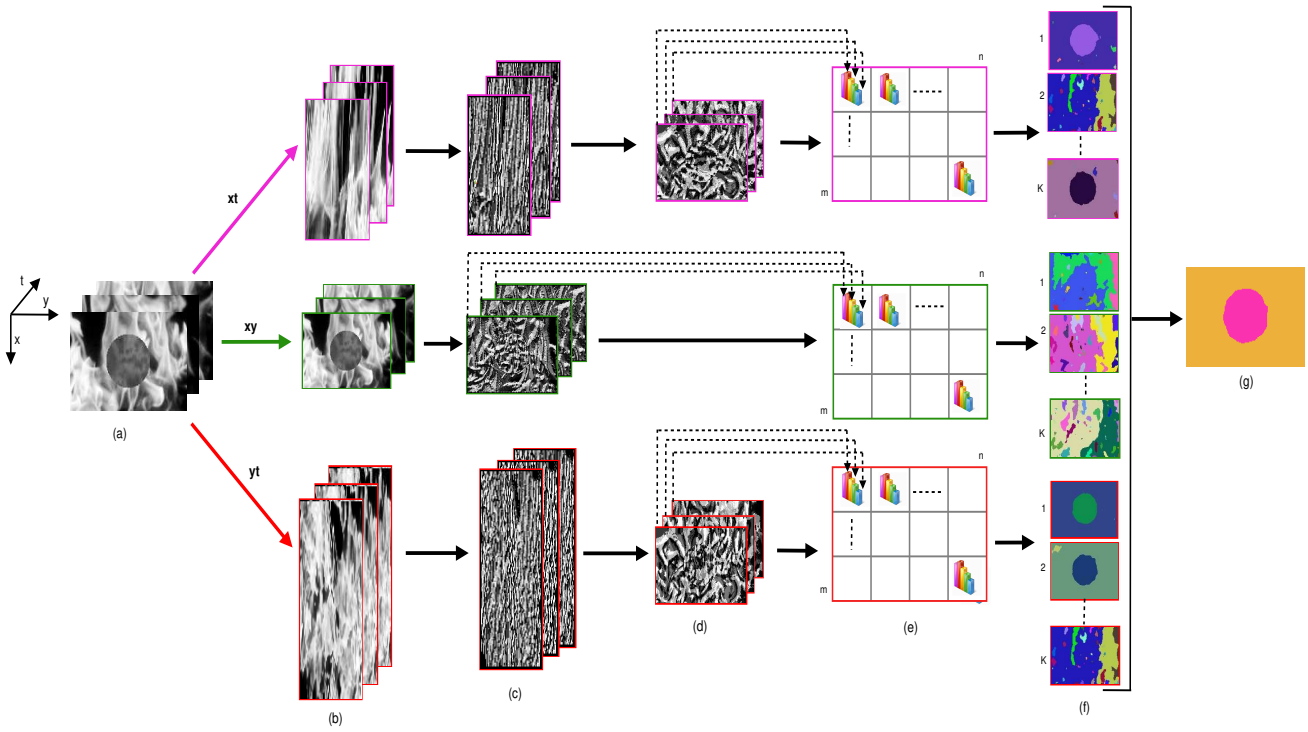

Figure 1. Proposed system overview. (a) Input video. (b) Slicing step . (c) LBP representation. (d) Projection of LBP frames on the xy plan. (e) Feature extraction and dimensionality reduction. (f) Clustering with k-means. (f) Final combined result.

segments in Φ_μ and nb_ν the number of segments in Φ_ν. Let now p_i be a particular pixel and the couple ($s_\mu^{<p_i>}$,$s_\nu^{<p_i>}$) be the two segments including this pixel, respectively in Φ_μ and Φ_ν. The LRE on this pixel p_i is the defined as follows:

$$\mathrm{LRE}(s_\mu, s_\nu, p_i) = \frac{|s_\mu^{<p_i>} \setminus s_\nu^{<p_i>}|}{|s_\mu^{<p_i>}|} \tag{2}$$

where $|X|$ denotes the cardinality of the set of pixels X and $\setminus$ represents the algebraic operator of difference. Particularly, a value of 1 means that the two regions overlap, in an inconsistent manner, on the contrary, an error of 0 expresses that the pixel is practically included in the refinement area [15]. A great way of forcing all local refinement to be in the same direction is to combine the LRE. By doing so, every pixel p_i must be computed twice, once in each sense, and in fact gives as result the so-called global consistency error (GCE):

$$\mathrm{GCE}^\star(\Phi_\mu, \Phi_\nu) = \frac{1}{2n}\left\{\sum_{i=1}^{n}\mathrm{LRE}(s_\mu, s_\nu, p_i) + \sum_{i=1}^{n}\mathrm{LRE}(s_\nu, s_\mu, p_i)\right\} \tag{3}$$

The GCE* value belongs in the interval of $[0, 1]$. On the one hand, a value of 0 expresses a maximum similarity between the two segmentationsΦ_μ. Φ_ν. On the other hand, the value of 1 represents a bad match or correspondence between the two segmentations to be compared.

Algorithm 1 Fusion algorithm

Mathematical notation:

$\overline{\mathrm{GCE}^\star}$	Mean GCE*
$\{\Phi_k\}_{k\leq J}$	Set of J segmentations to be fused
$\{b_j\}$	Set of superpixels $\in \{\Phi_k\}_{k\leq J}$
$\mathcal{E}$	Set of region labels in $\{\Phi_k\}_{k\leq J}$
$T_{\max}$	Maximal number of iterations
Φ_{best}	Fusion segmentation result

Input: $\{\Phi_k\}_{k\leq J}$

Output: Φ_{best}

A. Initialization:

$\Phi_I^{[0]} = \arg\min_{\Phi\in\{\Phi_k\}_{k\leq J}} \overline{\mathrm{GCE}^\star}(\Phi, \{\Phi_k\}_{k\leq J})$

B. Steepest Local Energy Descent:

while $p < T_{\max}$ **do**
  for each b_j superpixel $\in \{\Phi_k\}_{k\leq J}$ **do**
    • Draw a new label x according to the uniform distribution in the set $\mathcal{E}$
    • Let $\Phi_I^{[p],\mathbf{new}}$ the new segmentation map including b_j with the region label x
    • $\overline{\mathrm{GCE}^\star}_{\mathbf{new}} = \overline{\mathrm{GCE}^\star}(\Phi_I^{[p],\mathbf{new}}, \{\Phi_k\}_{k\leq J})$
    if $\overline{\mathrm{GCE}^\star}_{\mathbf{new}} < \overline{\mathrm{GCE}^\star}(\Phi_I^{[p]}, \{\Phi_k\}_{k\leq J})$ **then**
      • $\overline{\mathrm{GCE}^\star} = \overline{\mathrm{GCE}^\star}_{\mathbf{new}}$
      • $\Phi_I^{[p]} = \Phi_I^{[p],\mathbf{new}}$
      • $\Phi_{best} = \Phi_I^{[p]}$
    end if
  end for
  $p \leftarrow p + 1$
end while

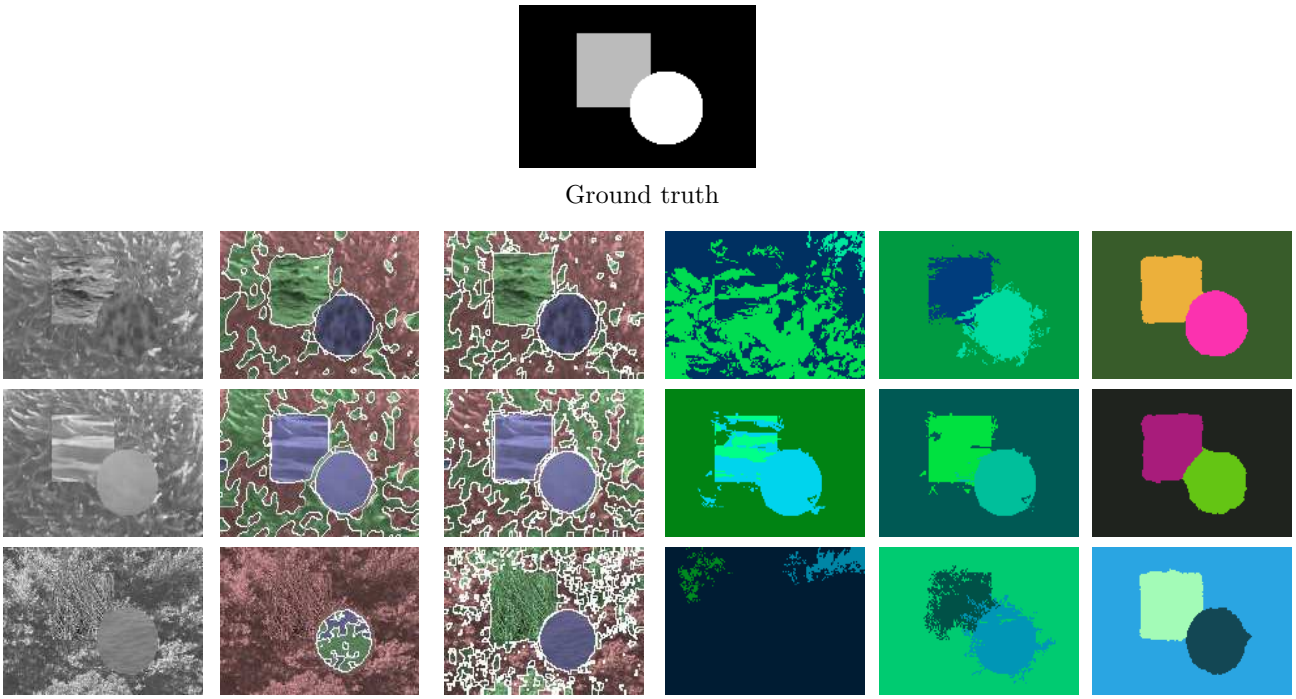

Figure 2. Examples of segmentation results obtained by our proposed method of three videos (with 3 labels) from the SynthDB dataset [22] compared to other algorithms.

2.4.2 Fusion

Let us assume now that $\{\Phi_k\}_{k\leq J} = \{\Phi_1, \Phi_2, \ldots, \Phi_J\}$ denotes the ensemble of J different segmentations to be combined. Note that J is equals to $3K$, where K represents the number of segmentations generated from each set of frames (see step (f) in Fig. 1). As stated in the section 1, our goal aim is to obtain a final improved segmentation result $\hat{\Phi}$ for the video sequence V. To find this refined segmentation result which represents a compromise or a consensus between the segmentations, an energy-based model is then built. Starting from an initial segmentation, this model aims to generate iteratively a segmentation solution which is as close as possible (with the GCE* considered distance) to all the other segmentations $\{\Phi_k\}_{k\leq J}$. In this framework, if Θ_n designates the set of all possible segmentations using n pixels, the consensus segmentation $\hat{\Phi}_{\overline{\text{GCE}}^\star}$ (optimal in the GCE* sense) is then straightforwardly defined as the minimizer of the $\overline{\text{GCE}}^\star$ function:

$$\hat{\Phi}_{\overline{\text{GCE}}^\star} = \arg\min_{\Phi\in\Theta_n} \overline{\text{GCE}}^\star\left(\Phi, \{\Phi_k\}_{k\leq J}\right) \qquad (4)$$

where Φ represents a segmentation that belongs the ensemble of all possible segmentations using n pixels. Our consensus model is formulated as a global optimization problem incorporating a nonlinear objective function. To minimize this energy function [see Eq.(4)], approximation approaches based on different optimization algorithms such as the exploration/selection/estimation (ESE) [16], the genetic algorithm or the simulated annealing [17] can be used. These algorithms are guaranteed to find the optimal solution, but with the drawback of a huge computational time. Another alternative adopted in this work is a semi-local optimization strategy based on the iterated conditional modes (ICM) method proposed by Besag [18] (i.e.; a Gauss-Seidel relaxation), where label of each region are updated one at a time [19]. In our case, this algorithm turned out to be both easy to implement, fast and efficient in terms of convergence properties. The different steps of the optimization process are summarized in Algorithm 1.

3. Experiments

We evaluate our method quantitatively on the synthetic video texture database (SynthDB) [22]. The SynthDB dataset [3] contains 299 8-bit grayscale video with the dimension of 160 $\times$ 110 $\times$ 60. Video sequences are split into three groups (99 videos with 2 labels, 100 videos with 3 labels, and 100 videos with 4 labels), and a common ground truth template is associated with each group. This dataset is very challenging, first because videos are grayscale, and also by the fact that textures exhibit very similar static ap-

Table 1. Comparison of the proposed method with other methods on the SynthDB dataset (PR index, higher is better).

ALGORITHMS	PERFORMANCE (Avg. PR)		
	99 videos 2 labels	100 videos 3 labels	100 videos 4 labels
GPCA [4] in [21]	0.515	0.477	0.526
DTM [22]	0.907	0.847	0.859
Color (Unsupervised) [8]	N/A	0.599	N/A
Color + motion (Unsupervised) [8]	N/A	0.727	N/A
Color + motion (Learned, logistic regression) [8]	N/A	0.771	N/A
Color+mouvment (Unsupervised) [8]	0.7113	0.608	0.612
Color+HoME+mouvment (Unsupervised) [8]	0.863	0.795	0.744
Baseline Init. (in [2])	0.600	0.684	0.704
Chen et al. [2]	0.924	0.884	0.855
-**Proposed method**-	**0.953**	**0.855**	**0.796**

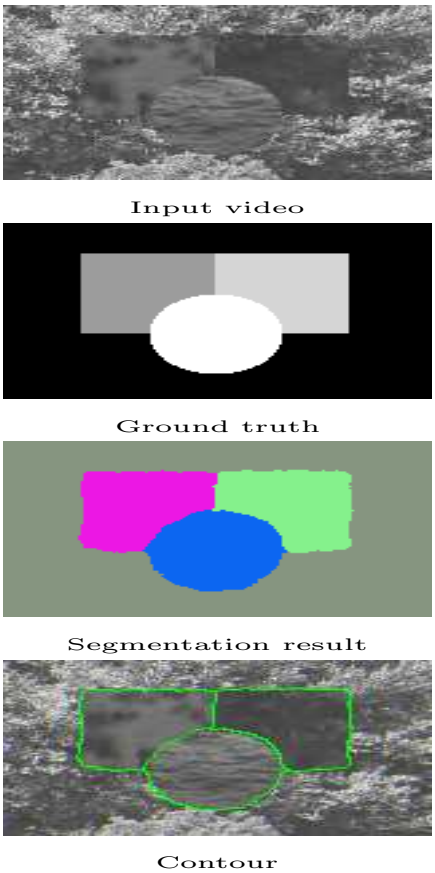

Figure 3. Example of segmentation result of a video with 4 labels from the SynthDB dataset [22].

pearance. Video segmentation performance is measured by the probabilistic rand (PR) index [20], which is widely used for evaluating the performances of related tasks. A score of one indicates a good result, otherwise, a score of zero indicates a bad segmentation. Table 1 shows that the result achieved by our unsupervised method outperforms to the state-of-the-art methods although it is not required any specific initialization step. For example, over the set of 99 videos (two labels) we obtain a good performance with an average PR equals to 0.953. Additionally, in Fig. 2, we present a qualitative comparison with other methods; layered dynamic textures (LDT)[23], dynamic texture model (DTM) [22], unsupervised and supervised (based learning metric) approaches proposed in [8]. The result of the proposed method, as shown in the sixth column, is clearly better than that of other methods. Also, in Fig. 3 we show a segmentation result of a video with 4 labels (or regions). To sum up, our method is simple, efficient and clearly has an advantage over complex, computationally demanding video segmentation models existing in the literature. Finally, it is worth mentioning that improvements can be made efficiently in our model by combining different features or using another fusion criteria better than the GCE.

4. Conclusion

A new approach for video segmentation with dynamic textures is proposed in this paper. Our method combine (based on a new geometric criterion) multiple and weak

[3]The synthetic video texture database is publicly accessible via this link: http://www.svcl.ucsd.edu/projects/motiondytex/

region-based segmentation maps to get a final better segmentation result. Experiments show that our results are comparable and even improve on state of the art methods using unsupervised and supervised strategy. Further studies, which take color video into account, will need to be undertaken. By doing so, future work will concentrate on the idea of combining different types of features with the local binary pattern (LBP) to more represent the color dynamic texture.

References

[1] F. Hajati, M. Tavakolian, S. Gheisari, Y. Gao, and A. S. Mian. Dynamic Texture Comparison Using Derivative Sparse Representation: Application to Video-Based Face Recognition. *IEEE Transactions on Human-Machine Systems*, 47(6):970–982, 2017.

[2] J. Chen, G. Zhao, M. Salo, E. Rahtu, and M. Pietikainen. Automatic Dynamic Texture Segmentation Using Local Descriptors and Optical Flow. *IEEE Transactions on Image Processing*, 22(1):326–339, 2013.

[3] G. Doretto, D. Cremers, P. Favaro, and S. Soatto. Dynamic texture segmentation. *In Proceedings of the 9th IEEE International Conference on Computer Vision (ICCV)*, pages 1236–1242, 2003.

[4] R. Vidal, and A. Ravichandran. Optical flow estimation and segmentation of multiple moving dynamic textures. *In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 516–521, 2005.

[5] K. Wattanachote, and T. K. Shih. Automatic Dynamic Texture Transformation Based on a New Motion Coherence Metric. *IEEE Transactions on Circuits and Systems for Video Technology*, 26(10):1805–1820, 2016.

[6] G. Farnebäk. Two-frame motion estimation based on polynomial expansion. *In Proceedings of the Scandinavian Conference on Image Analysis*, pages 363–370, 2003.

[7] T. M. Nguyen, and Q. J. Wu. An unsupervised feature selection dynamic mixture model for motion segmentation. *IEEE Transactions on Image Processing*, 23 (3):1210–1225, 2014.

[8] D. Teney, M. Brown, D. Kit, and P. Hall. Learning similarity metrics for dynamic scene segmentation. *In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2084–2093, 2015.

[9] V. Andrearczyk, and P. F. Whelan. Convolutional Neural Network on Three Orthogonal Planes for Dynamic Texture Classification.*Pattern Recognition*, 76:36–49, 2018.

[10] T. Ojala and M. Pietikäinen and T. Mäenpää. Multiresolution Gray-Scale and Rotation Invariant Texture Classification with Local Binary Patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(7):971–987, 2002.

[11] S. P. Lloyd. Least squares quantization in PCM. *IEEE Transactions on Information Theory*, 28(2):129–136, 1982.

[12] L. Khelifi, and M. Mignotte. A novel fusion approach based on the global consistency criterion to fusing multiple segmentations. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 47(9):2489–2502, 2017.

[13] A. Y. Yang, J. Wright, S. Sastry and Y. Ma. Unsupervised segmentation of natural images via lossy data compression. *Computer Vision and Image Understanding*, 110(2):212–225, 2008.

[14] L. Khelifi, and M. Mignotte. GCE-based model for the fusion of multiples color image segmentations. *In Proceedings of the 23rd IEEE International Conference on Image Processing (ICIP)*, pages 2574–2578, 2016.

[15] D. Martin, C. Fowlkes, D. Tal, and J. Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. *In Proceedings of the 8th International Conference on Computer Vision (ICCV)*, pages 416–423, 2001.

[16] F. Destrempes, M. Mignotte, and J.-F. Angers. A stochastic method for Bayesian estimation of hidden Markov models with application to a color model. *IEEE Transactions on Image Processing*, 14(8):1096–1108, 2005.

[17] L. Khelifi, I. Zidi, K. Zidi, and K. Ghedira. A hybrid approach based on multi-objective simulated annealing and tabu search to solve the dynamic dial a ride problem. *In Proceedings of the International Conference on Advanced Logistics and Transport (ICALT)*, pages 227–232, 2013

[18] J. Besag. On the statistical analysis of dirty pictures. *Journal of the Royal Statistical Society*, 48(3):259–302, 1986.

[19] L. khelifi, and M. Mignotte. Semantic image segmentation using the ICM algorithm. *In Proceedings of the 24th IEEE International Conference on Image Processing (ICIP)*, pages 3080–3084, 2017.

[20] R. Unnikrishnan, C. Pantofaru, and M. Hebert. A measure for objective evaluation of image segmentation algorithms. *In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 34–41, 2005.

[21] T. M. Nguyen, and Q. J. Wu . A Consensus Model for Motion Segmentation in Dynamic Scenes. *IEEE Transactions on Circuits and Systems for Video Technology*, 26(2):2240–2249, 2016.

[22] A. B. Chan, and N. Vasconcelos. Modeling clustering and segmenting video with mixtures of dynamic textures. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(5): 909–926, 2008.

[23] A. B. Chan, and N. Vasconcelos. Layered dynamic textures. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(10): 1862–1879, 2009.