

# Periodic Pattern Detection for Real-Time Application

Giovanni Puglisi<sup>1</sup> and Sebastiano Battiato<sup>1</sup>

Dipartimento di Matematica e Informatica  
University of Catania, Italy  
{puglisi,battiato}@dmi.unict.it

**Abstract.** Digital video stabilization approaches typically degrade their performances in presence of periodic patterns. Any kind of matching between consecutive frames is not usually able to work in presence of these kind of signals: the motion estimation engine is deceived and its performances degrade abruptly. In this paper we propose a fast fuzzy classifier able to recognize periodic and aperiodic pattern in the images that takes into account the peculiarities of digital video stabilization. Finally, the proposed classifier can be used as a filtering module in a block based video stabilization approach.

**Key words:** Video Stabilization, periodic pattern, fuzzy classifier

## 1 Introduction

In the last years video stabilization techniques have gained consensus, for they permit to obtain high quality and stable video footages even in non-optimal conditions. The best techniques, by using some mechanical tools, measure camera shake and then control the jitter acting on lens or on the CCD/CMOS sensor [1]. On the other hand, digital video stabilization techniques [2–6] make use only of information drawn from footage images and do not need any additional knowledge about camera physical motion.

Digital video stabilization systems have been widely investigated and several techniques have been proposed, with different issues and weak points. However in presence of regular or near regular texture, digital video stabilization approaches typically fail. These patterns, due to their periodicity, create multiple matching that degrade motion estimator performances. Even if some interesting approaches able to reliable find near regular texture have been recently developed [7, 8], they are pretty complex and cannot be applied in real-time digital video stabilization. In this paper we propose a fast fuzzy classifier able to find regular and low distorted near regular texture tacking into account video stabilization peculiarities.

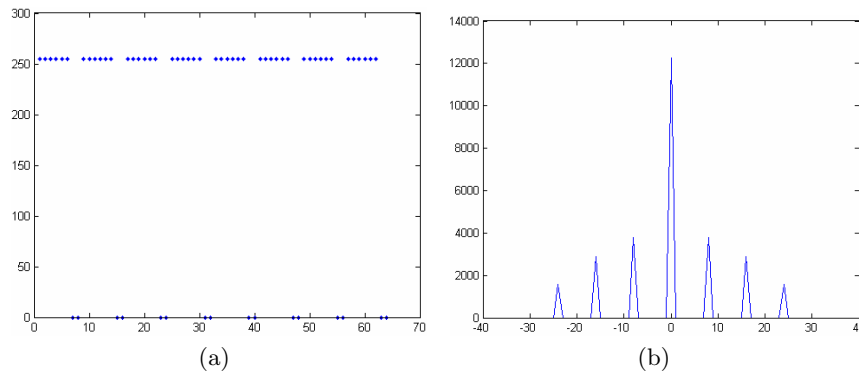
The rest of the paper is organized as follows. In Section 2 the analysis of the regular texture is performed. In Section 3 the classifier effectiveness applied to the video stabilization problem is shown whereas conclusions are summarized in Section 4.

## 2 Regular Texture Analysis

In real images we can find many regular and near regular texture such us: buildings, wallpapers, windows, floors, etc. In particular regular texture and low distorted near regular texture, due to the multiple matching candidates, typically create a lot of problems to motion estimation algorithms. On the contrary, in presence of high distorted near regular texture (very often created by perspective skewed patterns) video stabilization algorithms typically work in the correct way. Due to the limited number of samples in each selected patch and the spectral leakage that disperses frequencies over the entire spectrum a simple analysis (with proper thresholding) on Fourier peaks cannot be done. Also methods based on a simple analysis of bank filters (e.g., Gabor, etc.) are not able to properly detect the presence of such regions.

In this paper we propose a fuzzy classifier able to detect this kind of pattern in presence of some predefined constraints. It is based on Fourier domain analysis taking into account the following considerations (Fig. 1, 2):

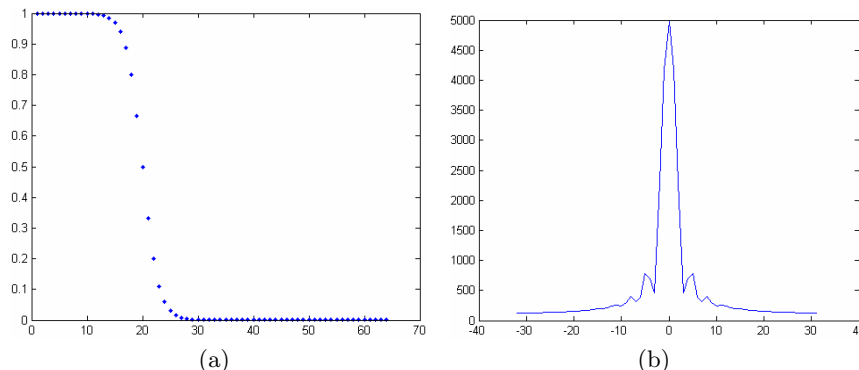
1. The highest Fourier spectrum values of a periodic signal have a greater distance from the axes origin than aperiodic signal values;
2. Fourier components of periodic signals typically have a lower density than aperiodic signal values.



**Fig. 1.** Example of 1-D periodic signal (a) with its spectrum (b).

The classifier makes use of the following two formulas:

$$distance = \frac{\sum_{i=-\frac{N}{2}}^{\frac{N}{2}-1} \sum_{j=-\frac{M}{2}}^{\frac{M}{2}-1} f_a(i, j) d(i, j)}{\sum_{i=-\frac{N}{2}}^{\frac{N}{2}-1} \sum_{j=-\frac{M}{2}}^{\frac{M}{2}-1} f_a(i, j) - f_a(0, 0)} \quad (1)$$



**Fig. 2.** Example of 1-D aperiodic signal (a) with its spectrum (b).

where  $d(i, j)$  is the Euclidean distance from the axes origin and  $f_a(i, j)$  is the Fourier spectrum component (defined in  $[-\frac{N}{2}, \frac{N}{2}-1] \times [-\frac{M}{2}, \frac{M}{2}-1]$ ) of a sequence of size  $N \times M$ .

$$density = \frac{\sum_{k=1}^{componentsNumber} neighbors(k)}{componentsNumber} \quad (2)$$

where *componentsNumber* represents the number of non-zero values of the Fourier spectrum and *neighbors(k)* the number of non-zero values close to the component  $k$ . The concept of closeness depends on the constraints of the particular application.

The noise contribution in the formulas described above has been reduced considering only the most important Fourier component values. We discard all the values less than 30% of the maximum without considering the DC (Direct Current) component.

### 3 Regular Texture Fuzzy Classifier

The formulas (1) and (2) can be effectively used as discriminant features in a simple fuzzy classifier with rules listed in Table 1.

The membership values of the fuzzy system have been derived considering the peculiarities of the particular application. We consider a video stabilization technique using a BM (block matching) estimation module with block size  $16 \times 16$  and search range  $\pm 16$  pixels. Block size defines the upper limit of periodic signal to be detected. The only periodic signals that must be taken into account, in this case, have a period less than 17 pixels.

A proper dataset containing both periodic and aperiodic images has been built by considering both synthetic and real texture downloaded from <sup>1 2 3</sup>. All

<sup>1</sup> <http://www.ux.uis.no/tranden/brodatz.html>

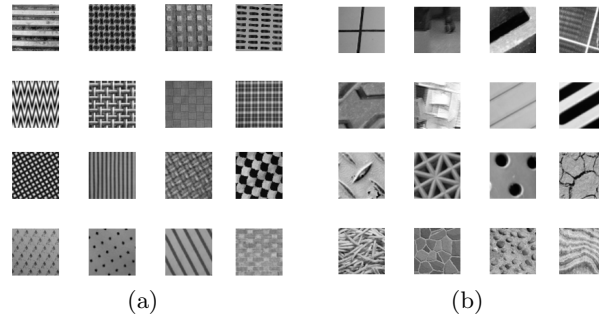
<sup>2</sup> <http://texturewarehouse.com/>

<sup>3</sup> <http://www.mayang.com/textures/>

the dataset (200 images) has been manually labeled in two classes: periodic and aperiodic. In the aperiodic group are also present corners, edges, regular texture with period greater than 16 pixels (our motion estimation algorithm, due to its local view considers them aperiodic) and irregular texture (Fig. 3).

**Table 1.** Fuzzy rules of the system.

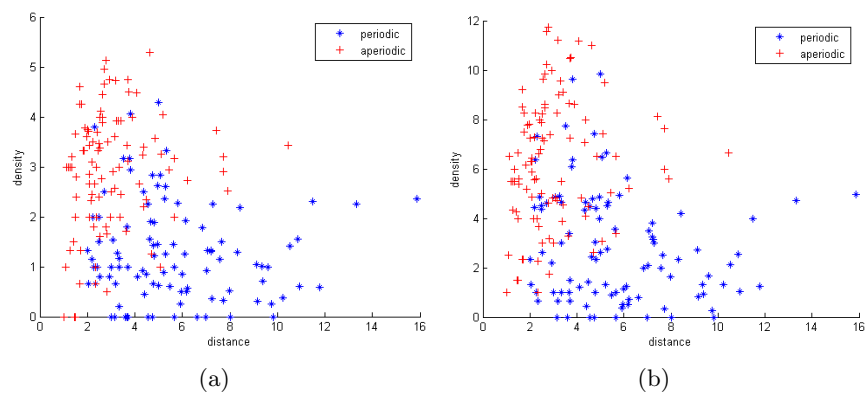
	<i>distance</i>		<i>density</i>		<i>periodicity</i>
if	<i>Low</i>	and	<i>Low</i>	then	<i>Low<sub>1</sub></i>
if	<i>Low</i>	and	<i>Medium</i>	then	<i>Low<sub>2</sub></i>
if	<i>Low</i>	and	<i>High</i>	then	<i>VeryLow</i>
if	<i>Medium</i>	and	<i>Low</i>	then	<i>High<sub>1</sub></i>
if	<i>Medium</i>	and	<i>Medium</i>	then	<i>Medium</i>
if	<i>Medium</i>	and	<i>High</i>	then	<i>Low<sub>3</sub></i>
if	<i>High</i>	and	<i>Low</i>	then	<i>VeryHigh</i>
if	<i>High</i>	and	<i>Medium</i>	then	<i>High<sub>2</sub></i>
if	<i>High</i>	and	<i>High</i>	then	<i>Low<sub>4</sub></i>



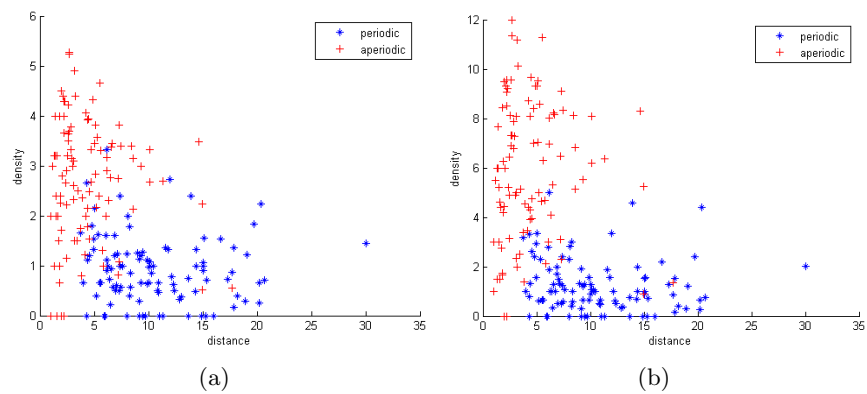
**Fig. 3.** Some periodic (a) and aperiodic (b) images belonging to our dataset (comprising images with period greater than 16 pixels).

In order to obtain better classification performances we have analyzed the distribution of periodic and aperiodic images by considering different image dimension and neighborhood size (Equation 2). As can be seen from Fig. 4, 5 image dimension equal to  $(64 \times 64)$  pixels and neighborhood size equal to 2 pixels provide the best distribution: periodic and aperiodic images are divided pretty well.

The training process, devoted to find membership parameters, has been performed using a continuous GA (genetic algorithm), an optimization and search technique based on the principle of genetics and natural selection. An initial population, usually randomly selected, of possible solutions evolves toward a



**Fig. 4.** Periodic and aperiodic images ( $32 \times 32$  pixels) in the features space just considering the neighborhood size equal to 1 (a) and 2 (b).

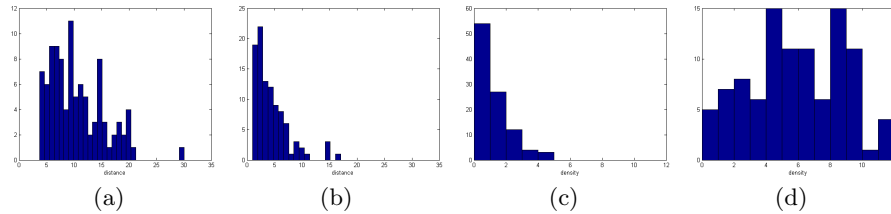


**Fig. 5.** Periodic and aperiodic images ( $64 \times 64$  pixels) in the features space just considering the neighborhood size equal to 1 (a) and 2 (b).

better solution. In each step some population elements are stochastically selected based on their fitness (the function to be optimized), and new elements are created through some techniques inspired by evolutionary biology (mutation, crossover). Genetic algorithms have found application in many fields [9]: computer science, engineering, economics, chemistry, physics, etc. Notice that for training simplicity we have considered a Sugeno fuzzy model.

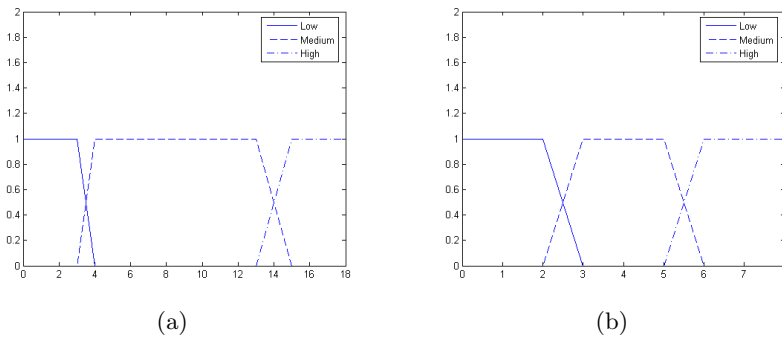
Genetic optimization is realized by using standard approaches in the field. In particular we have used default crossover and mutation algorithms provided by Genetic Toolbox functions of MATLAB 7. In order to speed-up the overall process, the initial population has been carefully defined. For our purposes the following considerations can be done (Fig. 6):

1. the minimum *distance* value of periodic patterns typically is greater than 4;
2. periodic signal typically have *density* values minor than 3.



**Fig. 6.** Histograms of *distance* (a, b) and *density* (c, d) features for periodic (a, c) and aperiodic signals (b, d).

The initial population elements have been derived through a gaussian random perturbation of the parameters shown in Fig. 7 and Tab. 2.



**Fig. 7.** Fuzzy membership functions for *distance* (a) and *density* (b) features.

**Table 2.** Fuzzy membership output values.

membership	<i>VeryLow</i>	<i>Low<sub>1</sub></i>	<i>Low<sub>2</sub></i>	<i>Low<sub>3</sub></i>	<i>Low<sub>4</sub></i>	<i>Medium</i>	<i>High<sub>1</sub></i>	<i>High<sub>2</sub></i>	<i>VeryHigh</i>
value	0	0.25	0.1	0.1	0.25	0.5	0.75	0.75	1

To validate our classifier we have performed a leave-one-out cross-validation. A single data is considered the validation dataset, and the remaining data the training dataset. Such procedure has been repeated until each data has been used as validation dataset.

For each input signal our fuzzy system produces a value belonging to [0-1] that is related to its degree of periodicity. In our case we choose as defuzzification strategy a simple thresholding process (threshold equal to 0.5). Table 3 reports the relative confusion matrix that confirms the robustness of the method for both classes, reaching an overall accuracy of 93%.

**Table 3.** Confusion matrix.

	periodic	aperiodic
periodic	95	5
aperiodic	9	91

The proposed classifier can be used as a filtering module in the video stabilization algorithms. Each region classified as periodic should be removed before starting the actual video stabilization process. In order to confirm the effectiveness of the classifier for this purpose we have compared the performances of a block based video stabilization approach [2] with and without periodic patterns removal step. As can be easily seen from Fig. 8, periodic patterns degrade [2] performances. On the contrary by simply using the filtering step proposed in this paper the video stabilization systems works pretty well.

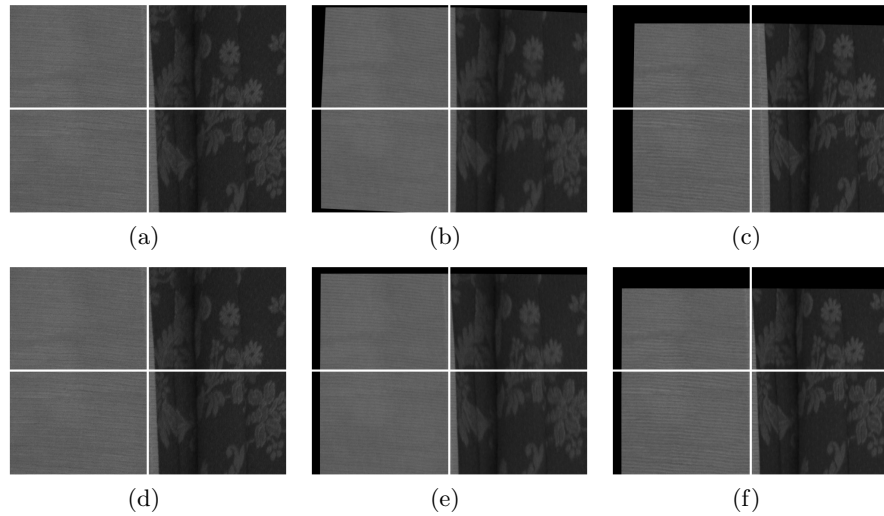
## 4 Conclusions

In this paper we have proposed a novel fast fuzzy classifier for low distorted near regular texture detection (and removal). For each input signal our system produces a value belonging to [0-1] that is related to its degree of periodicity. The classifier has been validated with a leave-one-out cross-validation obtaining an accuracy of 93%.

Future works will be devoted to extend this approach to other motion estimation constraints. Also feature extraction on DCT domain will be analyzed.

## References

1. Lukac, R.: Single-Sensor Imaging: Methods and Applications for Digital Cameras. CRC Press (2008)



**Fig. 8.** Stabilized images obtained by [2] with (d, e, f) and without (a, b, c) the periodic pattern removal step. The grid is overlaid for better visualization.

2. Battiato, S., Bruna, A.R., Puglisi, G.: A robust video stabilization system by adaptive motion vectors filtering. In: Proc. of ICME, Hannover (Germany) (June 2008)
3. Battiato, S., Gallo, G., Puglisi, G., Scellato, S.: SIFT features tracking for video stabilization. In: Proc. of ICIAP, Modena (Italy) (Sept. 2007) 825–830
4. Chang, J., Hu, W., Cheng, M., Chang, B.: Digital image translational and rotational motion stabilization using optical flow technique. *IEEE Transactions on Consumer Electronics* **48**(1) (February 2002)
5. Mercenaro, L., Vernazza, G., Regazzoni, C.: Image stabilization algorithms for video-surveillance application. In: Proc. of ICIP, Thessaloniki (Greece) (Oct. 2001)
6. Tico, M., Alenius, S., Vehvilainen, M.: Method of motion estimation for image stabilization. In: Proc. of IEEE ICASSP, Toulouse (France) (May 2006)
7. Hays, J.H., Leordeanu, M., Efros, A.A., Liu, Y.: Discovering texture regularity as a higher-order correspondence problem. In: Proc. of ECCV. (May 2006)
8. Liu, Y., Lin, W.C., Hays, J.H.: Near regular texture analysis and manipulation. *ACM Transactions on Graphics* **23**(3) (Aug. 2004) 368 – 376
9. R. L. Haupt, S.E.H.: *Practical Genetic Algorithms*. John Wiley & Sons, Hoboken, New Jersey, USA (2004)