Noise and Compression Robust Biological Features for Texture Classification

Gaëtan Martens, Chris Poppe, Peter Lambert, Rik Van de Walle

In: The Visual Computer, Volume 6, Numbers 6-8

http://www.springerlink.com/content/rm02107333k50846/

Noname manuscript No.
(will be inserted by the editor)

Noise and Compression Robust Biological Features for Texture Classification

Gaëtan Martens · Chris Poppe · Peter Lambert · Rik Van de Walle

Received: date / Accepted: date

Abstract Texture classification is an important aspect of many digital image processing applications such as surface inspection, content-based image retrieval, and biomedical image analysis. However, noise and compression artifacts in images cause problems for most texture analysis methods. This paper proposes the use of features based on the human visual system for texture classification using a semi-supervised, hierarchical approach. The texture feature consists of responses of cells which are found in the visual cortex of higher primates. Classification experiments on different texture libraries indicate that the proposed features obtain a very high classification near 97%. In contrast to other well-established texture analysis methods, the experiments indicate that the proposed features are more robust to various levels of speckle and Gaussian noise. Furthermore, we show that the classification rate of the textures using the presented biologically inspired features is hardly affected by image compression techniques.

Keywords Texture classification · Grating cell · Gabor · Noise · Image compression

1 Introduction

An important aspect of many image processing applications, such as surface inspection, image retrieval and medical image processing, is the classification and interpretation of textures. Texture classification involves deciding what texture class an observed sample belongs to. Although a plethora of texture analysis methods have been proposed in the literature, there are still some open issues where many applications struggle with. For most practical applications, noise is a frequent problem in texture characterization and causes difficulties for interpretation. Furthermore, different methods used to eliminate the noise, e.g., adaptive and non-adaptive filtering, eliminate some actual image data as well which results in the loss of texture information. Due to the high data volume of digital images and video, compression is a popular technique to reduce the size and to optimize the storage. However, the presence of artifacts caused by compression algorithms is an issue for texture analysis. Aschkenasy et al. [1] have investigated the effect of compression on texture analysis of echocardiographic images. They recommend the use of uncompressed or lossless compressed digital images in studies involving texture analysis since lossy compression affects texture parameters.

In the literature some approaches exist to deal with noise. For example, Iakovidis et al. [12] describe a fuzzy extension to the local binary pattern (LBP) operator [25] to deal with noise in images. An important drawback of their approach is that a parameter to control the degree of fuzziness has to be specified and an optimal value for this parameter is highly content-dependent. They also assume that the training and test images contain an equal amount of noise. Murino et al. [24] apply high-order statistics to create a feature vector. However, a selection algorithm is needed to reduce the high-dimensional data for classification. Fountain and Tan [9] used a multi-channel Gabor filter bank for classifying textures from the Brodatz album. By process-

Department of Electronics and Information Systems
Multimedia Lab, Ghent University – IBBT
Gaston Crommenlaan 8, bus 201
B-9050 Ghent-Ledeberg, Belgium
Tel.: +32 (0)93314959
Fax: +32 (0)93314896
E-mail: {gaetan.martens, chris.poppe, peter.lambert, rik.vandewalle}@ugent.be

ing an image using multiple resolution techniques, filter banks have the ability to decompose an image into relevant texture features that can be used to classify the textures accordingly. They found that their method was relatively robust to Gaussian noise by using a sufficient number of features. In computer vision the use of visual neuroscience has often been limited to a tuning of Gabor filter banks [6,7,15]. Little or no attention has been given to biological features of higher complexity. In 1992, Von der Heydt et al. [30] reported on the existence of a new type of orientation-selective neuron in the visual cortex of higher primates which they called grating cells. Grating cells respond vigorously to a grating of bars of appropriate orientation and periodicity, but they don't respond to bars that do not make part of a grating. The main purpose of grating cells in the human visual system (HVS) is a fast and reliable detection of periodic patterns of a certain orientation. The behavior of grating cells could be a vital property of a robust texture operator since they don't respond to non-texture features such as isolated pixels or edges. There exist some computational models of grating cells in the literature, e.g., by Kruizinga and Petkov [17], Lourens et al. [18], and Zhang et al. [31]. Despite these models, the application of grating cell features in the computer vision domain is, at the time of writing, still very limited. Zhang et al. apply their model in medical image retrieval, and, Ma and Doermann [19] use the model of Kruizinga and Petkov for classifying fonts of scanned document images. They have found that the grating cell operator can capture more texture differences between fonts than using isotropic Gabor filter responses. In previous work [21], we have used grating cell outputs for segmenting textured and outdoor scenery images and we obtained better results compared to the use of widely accepted Gabor filter outputs.

In this paper, we present the use of features inspired by the HVS for texture classification. Furthermore, we investigate the robustness against Gaussian and speckle noise and, against artifacts of a Joint Photographic Expert Group (JPEG) image compression algorithm. The remainder of the paper is structured as follows. Section 2 explains the computational model for calculating texture features and the relationship with the HVS. In Sect. 3, the filter bank design and the construction of our texture features is explained. Then in Sect. 4, we explain how the textures are classified using a hierarchical, semi-supervised neural network. In Sect. 5, we conduct classification experiments of the proposed texture features on two image sets and compare their results with other well-established texture analysis methods. The robustness of the proposed textures features against noise and image compression are tested. Finally, Sect. 6 ends this paper with conclusions.

2 Computational model

The computational model of the texture features we propose for classification is described in this section. At first, we take a closer look in Sect. 2.1 at the configuration of the Gabor filter which is at the basic level of our method. The model of Petkov and Kruizinga is briefly explained in Sect. 2.2 to compute enhanced grating cell responses. Finally, Sect. 2.3 considers the spatial smoothing of Gabor responses with regard to texture analysis.

2.1 Gabor filter functionality

In the spatial domain, a Gabor function is a Gaussian modulated by a sinusoid. To model the receptive fields of simple cells in the visual cortex, the real part of the following family of 2-dimensional Gabor filters are used as proposed by Daugman [8]:

$$g_{\lambda,\theta,\varphi}(x,y) = \frac{\exp\left(\left[\frac{-x'^2}{2\sigma_x^2} - \frac{\gamma^2 y'^2}{2\sigma_y^2}\right]\right)\cos\left(\frac{2\pi x'}{\lambda} + \varphi\right)}{2\pi\sigma_x\sigma_y} \quad (1)$$

where

$$\begin{cases} x' = x\cos\theta - y\sin\theta \\ y' = x\sin\theta + y\cos\theta. \end{cases}$$

The standard deviations σ_x and σ_y of the Gaussian factor determine the effective size of the surrounding of a pixel in which the summation takes place. A circular Gaussian is preferred so that there is a constant spatial extent in all directions, therefore $\sigma_x = \sigma_y (= \sigma)$. The parameter λ is the wavelength of the sinusoid, and the ratio σ/λ determines the bandwidth of the filter. Experiments indicate that the frequency bandwidth of simple cells is about one octave [28], thus $\sigma/\lambda \approx 0.56$. The spatial aspect ratio γ determines the eccentricity and herewith the eccentricity of the receptive field ellipse. According to Jones and Palmer [14], it has been found that γ varies in a limited range of $0.23 < \gamma < 0.92$ and is set to a constant value of 0.5. Further, the orientation of the filter is denoted by $\theta \in [0, \pi[$. This is the normal to the parallel lobes of the filter in the spatial-frequency domain, denoted by x' in equation (1). Finally, the phase offset φ affects the symmetry of the function. For $\varphi = 0$ or $\varphi = \pi$ the filter is symmetric while for $\varphi = \pi/2$ or $\varphi = -\pi/2$ the filter is anti-symmetric.

The response of the receptive field function of a simple cell tuned to orientation θ and frequency $1/\lambda$ to the

luminance channel of an input image $I(x,y)$ is then given by:

$$r_{\lambda,\theta,\varphi}(x,y) = \iint I(s,t) g_{\lambda,\theta,\varphi}(x-s, y-t) ds dt. \quad (2)$$

2.2 Enhanced grating cell operator

Grating cells respond to bar gratings of a given orientation and periodicity, but not to single bars. In order to better distinguish the salient texture-specific periodicities and to obtain an improved texture discrimination, an enhanced image $\overline{I}(x,y)$ is created by applying a histogram equalization to the original input image $I(x,y)$. Histogram equalization is a well-known technique that rescales the range of the pixel values to produce an image whose pixel values are more uniformly distributed which results in an image with a higher contrast. In previous work, we found that applying a histogram equalization increases the performance of texture segmentation when using grating cell outputs [20].

To model the non-linear behavior of the grating cells, we make use of the model of Kruizinga and Petkov [17]. This model first computes the output $s_{\lambda,\theta,\varphi}(x,y)$ to an input signal, here $\overline{I}(x,y)$, of a simple cell of the visual cortex tuned to a specific frequency $1/\lambda$ and orientation θ, by a non-linear stage which includes thresholding and contrast normalization:

$$s_{\lambda,\theta,\varphi}(x,y) = \begin{cases} 0 & \text{if } a_\lambda(x,y) = 0 \\ \chi\left(\frac{\frac{r_{\lambda,\theta,\varphi}(x,y)}{a_\lambda(x,y)} R}{\frac{r_{\lambda,\theta,\varphi}(x,y)}{a_\lambda(x,y)} + C} \right) & \text{otherwise,} \end{cases} \quad (3)$$

where R denotes the maximum response level, C is the semi-saturation constant, and $\chi(t) = t$ for $t \geq 0$ and $\chi(t) = 0$ for $t < 0$. The average gray value of the receptive field is given by

$$a_\lambda(x,y) = \iint \overline{I}(s,t) \exp \frac{(x-s)^2 + \gamma^2 (y-t)^2}{2\sigma^2} ds dt.$$

The simple cell outputs, given by equation (3), are then used to calculate the activity of grating subunits $q_{\lambda,\theta}$ which are summed up together to obtain a grating cell response:

$$q_{\lambda,\theta}(x,y) = \begin{cases} 1 \text{ if } \forall n, M_{\lambda,\theta,n}(x,y) \geq \rho N_{\lambda,\theta}(x,y) \\ 0 \text{ if } \exists n, M_{\lambda,\theta,n}(x,y) < \rho N_{\lambda,\theta}(x,y) \end{cases} \quad (4)$$

where $0 < \rho < 1$ is a threshold value in the proximity of 1, e.g., $\rho = 0.9$. The quantities $M_{\lambda,\theta,n}$ and $N_{\lambda,\theta}$ for $\varphi_n = 0$, $n \in \{-3,-1,1\}$ and $\varphi_n = \pi$ for $n \in \{-2,0,2\}$ are given by:

$$\begin{cases} M_{\lambda,\theta,n}(x,y) = \max_{(u,v)} \{ s_{\lambda,\theta,\varphi_n}(u,v) \} \\ N_{\lambda,\theta}(x,y) = \max \{ M_{\lambda,\theta,n}(x,y) | n = -3,-2,..2 \} \end{cases}$$

Finally, u and v satisfy the condition:

$$\begin{cases} n\frac{\lambda}{2} \cos\theta \leq u - x < (n+1)\frac{\lambda}{2} \cos\theta \\ n\frac{\lambda}{2} \sin\theta \leq v - y < (n+1)\frac{\lambda}{2} \sin\theta. \end{cases}$$

A grating subunit will thus be activated ($q_{\lambda,\theta}(x,y) = 1$) if for the preferred orientation θ and spatial-frequency $1/\lambda$, the receptive field function $r_{\lambda,\theta,\varphi_n}$ is alternately activated in intervals of length $\lambda/2$ for $n = -3,-2,..,2$ and this along a line segment of length 3λ centered on point (x,y). This is the case if at least 3 parallel bars with spacing λ and orientation θ of the normal to them are encountered.

In the final stage, the output of the enhanced grating cell operator $\overline{w}_{\lambda,\theta}$ with frequency $1/\lambda$ and orientation θ to the normal of the grating is computed by a weighted summation of the responses of the subunits $q_{\lambda,\theta}$. The operator is made symmetric by considering the opposite direction $\theta + \pi$:

$$\overline{w}_{\lambda,\theta}(x,y) = \iint W(x,y) \left(q_{\lambda,\theta}(s,t) + q_{\lambda,\theta+\pi}(s,t) \right) ds dt \quad (5)$$

where $W(x,y) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(x-s)^2 + (y-t)^2}{2(5\sigma)^2} \right)$.

2.3 Spatial smoothing

Textures which do not have sufficiently narrow bandwidths may suffer from leakage. The effects of leakage can be reduced by post-filtering the channel amplitudes with Gaussian filters having the same shape as the corresponding channel filters but greater spatial extents. Therefore, smoothed Gabor responses are known to improve the performance for texture analysis [3]. There exists a physiological reason for utilizing smoothing since it mimics characteristics of the HVS. Hall and Hall [10] describe the existence of sustained channels in the visual system, indicating that the HVS not only considers pixels in the field of view, but also pixels in the vicinity.

The spatially smoothed Gabor responses we use, are obtained by convolving symmetric Gabor responses with a Gaussian with standard deviation $\sigma' = 2\sigma$:

$$\tilde{r}_{\lambda,\theta} = [r_{\lambda,\theta,0} * gauss](x,y) \quad (6)$$

where

$$gauss(x,y) = \frac{1}{2\pi\sigma'^2} \exp\left(-\frac{x^2 + y^2}{2\sigma'^2} \right).$$

3 Texture features

An important aspect for implementing a Gabor filter, is the size of the neighborhood window to compute the filter's response. In our model, the window size

k is related to the standard deviation of the Gaussian (and thus to the bandwidth of the filter), i.e., $k = nint(2\sigma) + 1 \approx nint(1.12\lambda + 1)$ pixels, where $nint$ denotes the nearest integer function. Consequently, the size of the window changes according to the periodicity of the texture: a smaller window for small periodicities, and a larger window for the larger periodicities.

For texture classification, the texture features we propose are composed as follows. For input images of size 512×512, the center frequencies for the filter banks are set to $\sqrt{2}, 2\sqrt{2}, 4\sqrt{2}, 8\sqrt{2}$, and $16\sqrt{2}$ cycles per image. Further, we use 8 orientations (θ=0, $\frac{\pi}{8}$,..,$\frac{7\pi}{8}$) to obtain a selectivity of 22.5 degrees.

Since $\tilde{r}_{\lambda,\theta}$ has a different range than $\overline{w}_{\lambda,\theta}$, a vector normalization is applied otherwise, larger values tend to dominate the other values during classification. Concatenating the normalized enhanced grating cell responses $\overline{w}'_{\lambda,\theta}$ and the normalized smoothed Gabor filter responses $\tilde{r}'_{\lambda,\theta}$ to a pixel (x_i, y_j), results in an 80-dimensional feature vector $v(x_i, y_j)$:

$$v(x_i, y_j) = \left\{\overline{w}'_{\lambda_k,\theta_l}(x_i, y_j), \tilde{r}'_{\lambda_k,\theta_l}(x_i, y_j)\right\} \quad (7)$$

for $k = 0..4$ and $l = 0..7$.

4 Semi-supervised Classification

To classify the extracted feature vectors v, we make use of a hierarchical variant of the Self-Organizing Map (SOM). The SOM is a single layer artificial neural network that simulates the process of unsupervised self-organization with a simple, yet effective numerical algorithm [16]. There exists a lot of neurophysiologic evidence to support the idea that the SOM captures some of the fundamental processing principles of the human (both visual and auditory) cortex. An important property of the SOM is that it clusters similar data vectors and projects dissimilar ones far from each other on the map. A SOM includes a grid of nodes and each node is associated with a parametric real vector, called the model vector. For a given input, the model vectors are updated according to the following rule: (i) find the best matching unit (BMU) using a predefined metric (which is usually the Euclidean distance), and (ii) change the model vectors in a neighborhood of the BMU (the size of the neighborhood is a decreasing function of time). At first, a 2-dimensional SOM of a predefined size is trained with labeled data. A result of this training process is that pixels belonging to the same texture are assigned to the same or adjacent nodes.

However, we have experienced that the number of misclassifications increases with the number of textures in the training data. A way to reduce these errors, is by increasing the dimensions of the SOM. Though, a larger map will also increase the computation time. In addition, a larger SOM won't prevent that some nodes are 'contaminated', i.e. that a node is associated with training data from different textures. For example, when two textures are relatively similar to each other compared to several other textures in the training data, it is possible that these two textures are not distinguished by the SOM. To tackle this issue, we employ a hierarchical approach utilizing the labels of the training data as some means of supervision. The training vectors associated with a contaminated node are then used to train a new, smaller SOM. This process is iteratively repeated until a certain stopping criterion is reached or no progress in the classification is obtained. Suppose that a node N is related to a set V of training vectors v of k classes c_i, $i = 0..k-1$: $N \leftarrow V = \{v_{c_0,1}..v_{c_0,m_0}, ..v_{c_j,1}..v_{c_{k-1},m_{k-1}}\}$ where m_i denotes the number of training vectors of texture c_i assigned to N. The stopping criterion is defined by a threshold $0 < \tau \leq 1$:

$$\frac{\max_{i=0..k-1}\{m_i\}}{\sum_{i=0}^{k-1} m_i} \geq \tau \quad (8)$$

If (8) isn't satisfied, V is used to train a new SOM. This process is then repeated for the nodes of the resulting SOM. The class of an unknown test sample is easily obtained by calculating its BMU. If the BMU is an empty node (no vectors were assigned to this node during the training phase), the label of its closest node is used. The parameter τ is empirically set to 0.95 and the dimensions of the SOM for K texture classes are chosen as follows: 4×4 for $K = 2$ or 3, 8×8 for $K = 4$, 10×10 for $K = 5$, 15×15 for $K = 6$ and 20×20 for $K \geq 7$.

5 Classification Experiments

We use the SOM Toolbox [29] to create the SOMs, and the Euclidean distance is employed as distance metric. The performance and robustness of the SOM-based classification with the proposed texture features is analyzed using two different data sets: 10 textures from the Brodatz [4] (D6, D9, D12, D15, D19, D38, D68, D84, D94, D104), and 10 textures from the Vistex album [27] (Grass.0001, Bark.0012, Brick.0000, Metal.0002, Fabric.0005, Fabric.0015, Food.0006, Leaves.0008, Sand.0001, Water.0005). From the 512×512 pixels images, we crop an area of 512×452 pixels and use it as training data. The remaining 512×60 pixels are then used as test data for classification. The texture features are extracted from patches of 4×4 pixels as a compromise between the computation time and detail. This results in a total of 1920 test features per texture.

In order to investigate the robustness of the proposed features v, we add zero mean speckle and Gaussian noise of different variances to the test images using the Matlab Image Processing Toolbox [22], and try to classify these noisy images using the (noise free) training data. In the last experiment, we apply JPEG compression of different quality levels using a compression algorithm of the Independent JPEG Group (IJG) [13] on the color Vistex textures and, classify them using the uncompressed training data. We also tackle these classification problems using Gaussian Markov Random Fields (GMRF) [5], multi-scale LBP [26], and fuzzy LBP [12]. The implementation of the GMRF features is obtained from the MeasTex site [23]. The 73-dimensional GMRF feature vectors are computed using the standard symmetric masks and are obtained by concatenating feature vectors of GMRF models of order 1 to 7 (GMRF_{1-7}). The multi-scale LBP, i.e. $LBP^{riu2}_{(8,1)+(16,2.4)}$, are uniform and rotation invariant and the 2-dimensional co-occurrence LBP histograms are classified using a non-parametric L-statistic [26]. The fuzzy LBP (FLBP) are calculated in a 3×3 neighborhood ($P = 8$ pixels and radius $R = 1$ pixel) with fuzzyfication parameter $T = 75$ as proposed in [12]. The resulting histograms are also classified using the L-statistic.

5.1 Experiment 1

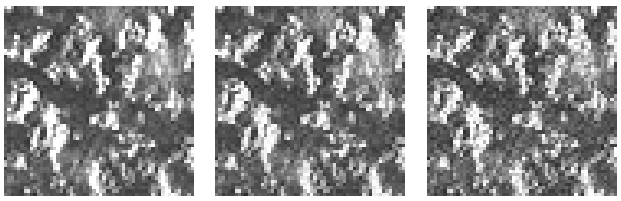
At first, the classification performance of the different texture features is separately tested on the two data sets using the noise free and uncompressed training data. As can be seen in Tab. 1, the proposed texture features and the multi-scale LBP attain the best classification performance. Also the GMRF feature vectors acquire a reasonable classification rate, in contrast to the FLBP that obtain a rate near 40% due to the small spatial support area (3×3 pixels).

Table 1 Classification rate (%) of noise free and uncompressed textures.

	proposed	GMRF_{1-7}	LBP^{riu2}	FLBP
Brodatz	97.8	81.5	97.7	39.0
Vistex	96.9	85.6	96.9	39.3

5.2 Experiment 2

In this experiment, we add zero mean speckle noise with different variances (0.0025, 0.005, 0.01, 0.02, 0.03, and 0.04) to the training data. The noisy textures are then classified using the neural network that is trained with the noise free and uncompressed texture data. Speckle noise is a random, granular noise, and the texture of the observed speckle pattern does not correspond to underlying structure. Although hardly visible at first sight for the human eye in the images (see Fig. 1), the amount of speckle noise has actually a strong influence on the classification rates. As can be seen in Fig. 2, the speckle noise affects the classification rates of all features, but the proposed texture features clearly obtain the best classification results and perform at least 25% better than the multi-scale LBP for the highest variance of the noise. For both the Brodatz and Vistex textures, the performance of the multi-scale LBP and the GMRF features drops as the variance of the noise increases. The classification rate of the FLBP slightly decreases, and, generally obtains a low performance. This not only due to the small spatial support of FLBP, but also to the fact that the classifier is trained with noise free data, while in the experiments of Iakovidis et al. [12] the training data also contain noise.



(a) (b) (c)

Fig. 1 Detail of the Vistex Metal.0002 texture (a) with zero mean speckle noise and variance (b) 0.01, and (c) 0.04.

5.3 Experiment 3

Similarly to previous experiment, zero mean Gaussian noise of different variances (0.0025, 0.005, 0.01, 0.02, 0.03, and 0.04) is now added to the test textures. As exemplified in Fig. 3, the visual quality of the textures is highly affected. Figure 4 plots the classification rate of the texture samples with the Gaussian noise. As can be seen, the proposed features obtain the best classification rate while the other methods struggle with the noise. The performance of the FLBP slightly decreases as the variance of the Gaussian noise increases, while the classification rate of the GMRF and the multi-scale LBP steadily drops.

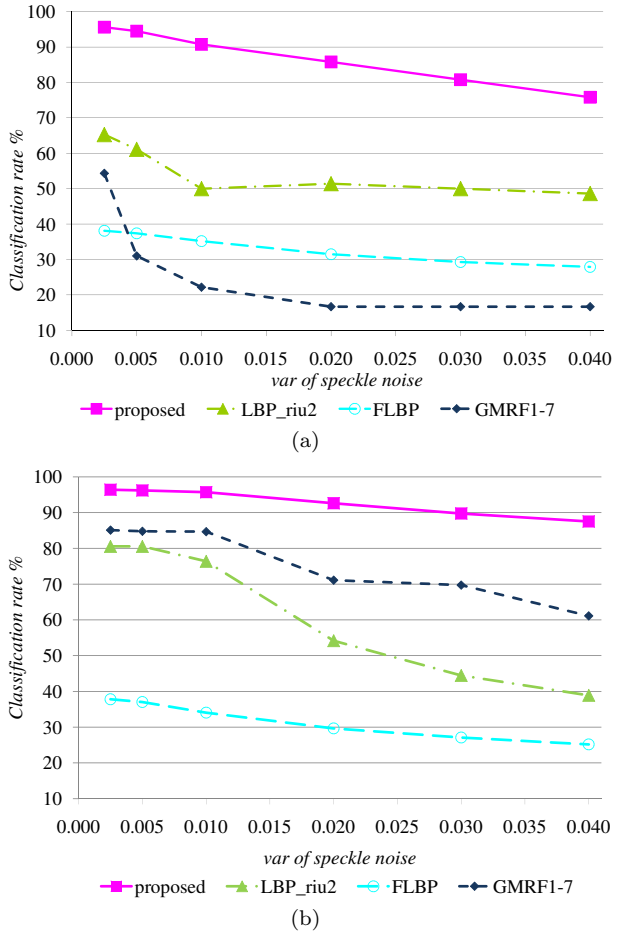


Fig. 2 Classification rate of (a) Brodatz and (b) Vistex textures in function of the variance of zero mean speckle noise.

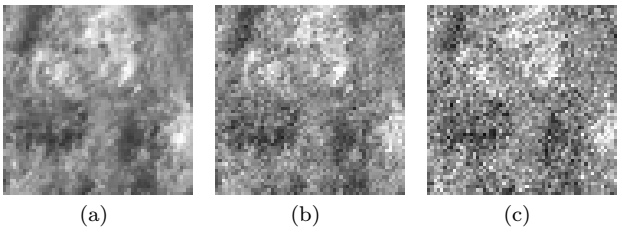


Fig. 3 (a) Detail of the Vistex Fabric.0015 texture, with zero mean Gaussian noise and variance (b) 0.01, and (c) 0.04.

5.4 Experiment 4

The robustness against JPEG compression is tested in this experiment. There is no direct measure of the degree of image distortion introduced by JPEG compression. We use the so-called 'quality levels' proposed by IJG, which range from 0 (lowest quality) to 100 (highest quality). Therefore, the Vistex color test images are compressed at different quality levels. Since little or no artifacts appear by compressing gray-scale images (JPEG compresses hue data more heavily than brightness data), the Brodatz pictures are not used in this experiment. The lower the quality level, the more compression artifacts that will arise and the high spatial-frequencies of the texture are more affected as exemplified in Fig. 5 As plotted in Fig. 6, it is obvious that

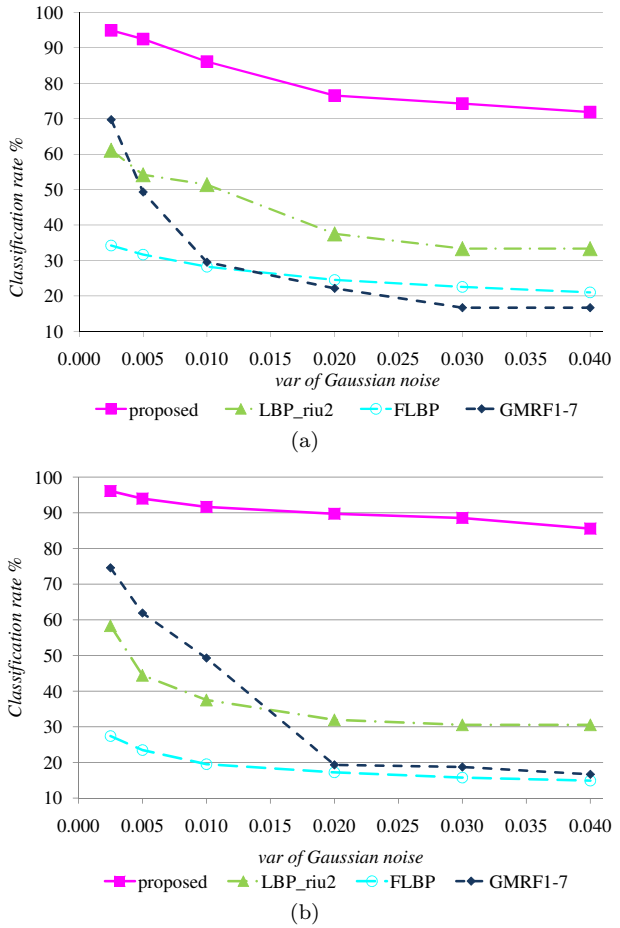


Fig. 4 Classification rate of (a) Brodatz and (b) Vistex textures in function of the variance of zero mean Gaussian noise.

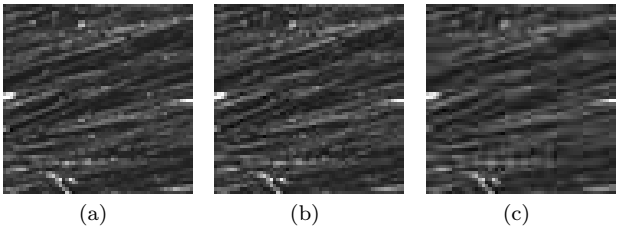


Fig. 5 (a) Detail of the Fabric.0005 texture, compressed at quality level (b) 75 and (c) 15.

the proposed features are hardly affected by the JPEG compression. The multi-scale LBP can keep up with the proposed features, but for quality levels lower than 75, the classification rate steadily drops and attains for

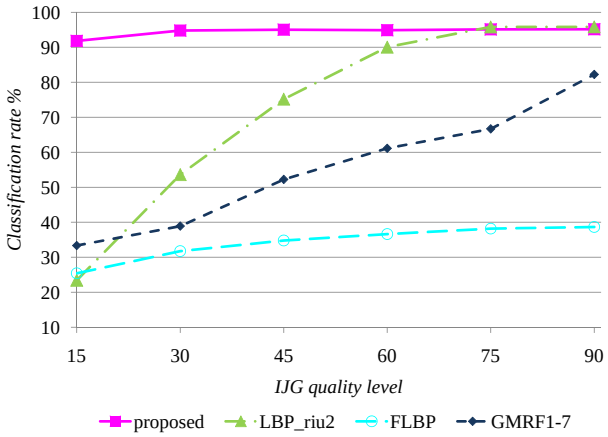


Fig. 6 Classification rate of JPEG compressed Vistex textures in function of the IJG quality levels.

quality level 15 a lower classification rate than FLBP or GMRF. Also, the already low classification rate of the FLBP drops for quality levels lower than 75. The classification rate of the GMRF starts to decrease from quality level 75.

6 Conclusions

This paper presented the use of HVS-inspired features for semi-supervised texture classification using hierarchical SOMs. Experiments conducted on textures taken from the Brodatz and Vistex album indicate that the proposed features obtain very high classification rates near 97%, and can thus compete with well-established methods like multi-scale LBP.

However, noise is a frequent problem in many practical applications and causes difficulties for texture interpretation. Therefore, we have investigated the robustness of texture analysis methods against various levels of speckle and Gaussian noise. We added zero mean speckle and Gaussian noise with different variances to the test data and, we classified these data using the noise-free training data. We have observed that the classification rate of the proposed features is less affected by the various amounts of noise and achieves better classification rates than the other methods. The proposed HVS-inspired features clearly obtain the best classifications results over all test sets.

Since JPEG image compression is a popular technique to reduce the size of image data, we have also investigated the impact of the compression rate on texture classification. Therefore, we applied JPEG compression at various quality levels on the test images and we observed that the proposed features are very robust to JPEG compression. We noticed that the classification rate of the other considered texture features starts to decrease as the quality level of the JPEG compression decreases, while it has hardly an effect on the classification rate of the proposed features.

It is obvious that our approach which is based on a filter bank can deal better with the information loss introduced by noise or image compression algorithms than methods like GMRF and LBP since the latter methods are more sensible to local changes of pixel values. To conclude, the proposed HVS-inspired features are robust to speckle and Gaussian noise, and, the classification rate is hardly affected by JPEG image compression.

Acknowledgment

The research activities as described in this paper were funded by Ghent University, the Interdisciplinary Institute for Broadband Technology (IBBT), the Institute for the Promotion of Innovation by Science and Technology in Flanders (IWT), the Fund for Scientific Research-Flanders (FWO-Flanders), and the European Union.

References

1. S.V. Aschkenasy, J. Muntwyler, B. van der Loo, E. Oechslin and R. Jenni, Texture analysis in digitally-acquired echocardiographic images: The effect of JPEG compression and video storage, Ultrasound in Medicine & Biology, 31 (3), pp. 361–366 (2005)
2. J. Beck, A. Sutter, and A. Ivry, Spatial Frequency Channels and Perceptual Grouping in Texture Segregation, Computer Vision Graphics Image Process, 37, pp. 299–352 (1987)
3. A.C. Bovik, M. Clark, and W.S. Geisler, Multichannel Texture Analysis Using Localized Spatial Filters, IEEE Transactions on Pattern Analysis and Machine Intelligence, 12 (1), pp. 55–73 (1990)
4. P. Brodatz, Textures: A Photographic Album for Artists and Designers, Dover, New York (1966)
5. R. Chellappa and S. Chatterjee, Classification of Textures Usinge Gaussian Markov Random Field, IEEE Trans Acoust Speech Sig Process, 33, pp. 959–963 (1985)
6. M. Clark and A. Bovik, Texture Segmentation Using Gabor Modulation/Demodulation, Patt Recogn Let, 6, pp. 261–267 (1987)
7. D.A. Clausi, M.E. Jernigan, Designing Gabor Filters for Optimal Texture Separability, Pattern Recognition, 33 (11), pp. 1835–1849 (2000)
8. J.G. Daugman, Uncertainty relation for resolution in space, spatial frequency, and orientation optimization by two-dimensional visual cortical filters. J. Opt. Soc. Am., 2, pp. 1160-1169 (1985)
9. S.R. Fountain and T.N. Tan, Extraction of noise robust rotation invariant texture features via multichannel filtering, Proceedings of the 1997 International Conference on Image Processing, 3, pp. 197–200 (1997)
10. C.F. Hall, E.L. Hall, A nonlinear model for the spatial characteristics of the human visual system, IEEE Trans. Systems Man Cybern., 7 (3), pp. 161–170 (1977)

11. R.M. Haralick, K. Shanmugam, and I. Dinstein, Textural Features for Image Classification, IEEE Trans. Syst. Man. Cybern, 3 (6), pp. 610–621 (1973)
12. D.K. Iakovidis, E. Keramidas, D. Maroulis, Fuzzy Local Binary Patterns for Ultrasound Texture Characterization, Proc. Image Analysis and Recognition, 5th International Conference (ICIAR 2008), Lecture Notes in Computer Science, vol. 5112, Springer, pp. 750–759 (2008)
13. Independent JPEG Group, http://www.ijg.org/
14. J. Jones and A. Palmer, An Evaluation of the Two Dimensional Gabor Filter Model of Simple Receptive Fields in Cat Striate Cortex, J. of Neurophysiology, 58, pp. 1233–1258 (1987)
15. N.N. Kachouie and J. Alirezaie, Optimized multichannel filter bank with flat frequency response for texture segmentation, Journal on Applied Signal Processing, 12, pp. 1834–1844 (2005)
16. T. Kohonen, Self-organizing Maps, Springer-Verlag, Berlin, Germany (1997)
17. P. Kruizinga and N. Petkov, Nonlinear operator for blob texture segmentation, in: Proc. NSIP'99, IEEE Workshop on Nonlinear Signal Processing, 2, A.S. Cetin, et al. (Eds.), pp. 881–885 (1999)
18. T. Lourens, E. Barakova, H.G. Okuno, and H. Tsujino, A computational model of monkey cortical grating cells, Biological Cybernetics, 92 (1), pp. 61–70 (2005)
19. H. Ma and D.S. Doermann, Font identification using the grating cell texture operator, in: Proceedings of Document Recognition and Retrieval XII, pp. 148–156 (2005)
20. G. Martens, C. Poppe, R. Van de Walle, Enhanced Grating Cell Features for Unsupervised Texture Segmentation, Performance Evaluation for Computer Vision: 31ste AAPR/OAGM Workshop 2007, Osterreichische Computer Gesellschaft, Performance Evaluation for Computer Vision, pp. 9–16 (2007)
21. G. Martens, C. Poppe, P. Lambert, R. Van de Walle, Unsupervised Texture Segmentation and Labeling Using Biologically Inspired Features, in: Proceedings of the 2008 IEEE 10th workshop on Multimedia Signal Processing, IEEE Signal Processing Society, pp. 159–164 (2008)
22. Matlab Image Processing Toolbox 5.1, see: http://www.mathworks.com/products/image/
23. MeasTex Image Texture Database and Test Suite, Version 1.1, see: http://www.texturesynthesis.com/meastex/meastex.html.
24. V. Murino, C. Ottonello, S. Pagnan, Noisy texture classification: A higher-order statistics approach. Pattern Recognition, 31 (4), pp. 383–393 (1998)
25. T. Ojala, M. Pietikäinen, D. Harwood: A comparative study of texture measures with classification based on feature distributions, Pattern Recognition, 29 (1), pp. 51-59 (1996)
26. T. Ojala, M. Pietikäinen and T. Mäenpää, Multiresolution gray-scale and rotation invariant texture analysis with local binary patterns, IEEE Transactions on Pattern Analysis and Machine Intelligence, 24 (7), pp 971–987 (2002)
27. R.W. Picard and T.P. Minka, Vision Texture for Annotation, Multimedia Systems, 3 (1), pp. 3–14 (1995)
28. D.A. Pollen and S.F. Ronner, Visual cortical neurons as localized spatial frequency filters, IEEE Trans. Systs, Man, and Cybern, 13 (5), pp. 907–916 (1983)
29. J. Vesanto, J. Himberg, E. Alhoniemi and J. Parhankangas, Self-Organizing Map in Matlab: the SOM Toolbox, in: Proceedings of the Matlab DSP Conference, pp. 35–40 (2000)
30. R. von der Heydt , E. Peterhans, and M.R. Dürsteler, Periodic-pattern-selective cells in monkey visual cortex, Journal of Neuroscience, 12 (4), pp. 1416-1434 (1992)
31. G. Zhang, Z.M. Ma, Z. Cai and H. Wang, Texture Analysis Using Modified Computation Model of Grating Cells in Content-Based Medical Image Retrieval, Lecture Notes in Computer Science, vol. 4989, Springer, pp. 125–132 (2008)