Wavelet-Based Texture Retrieval Using a Mixture of Generalized Gaussian Distributions

Mohand Saïd Allili
Université du Québec en Outaouais
Département d'Informatique et d'Ingénierie,
101, Rue St-Jean-Bosco, Local: B-2022,
Gatineau, Québec, J8X 3X7, Canada.
Email: mohandsaid.allili@uqo.ca.

***Abstract*—In this paper, we address the texture retrieval problem using wavelet distribution. We propose a new statistical scheme to represent the marginal distribution of the wavelet coefficients using a mixture of generalized Gaussian distributions (MoGG). The MoGG allows to capture a wide range of histogram shapes, which provides a better description of texture and enhances texture discrimination. We propose a similarity measurement based on Kullback-Leibler distance (KLD), which is calculated using MCMC Metropolis-Hastings sampling algorithm. We show that our approach yields better texture retrieval results than previous methods using only a single probability density function (pdf) for wavelet representation, or texture energy distribution.**

***Keywords*-avelet decomposition, Mixture of Generalized Gaussians, texture image retrieval, KLD.**

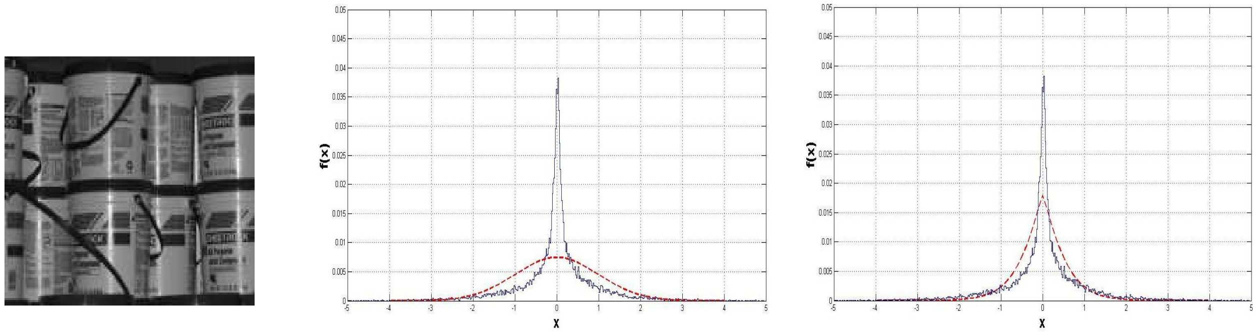
I. Introduction

Wavelet decomposition is a very useful tool for image and video representation. It allows to represent an image/video signal in independent frequency bands, where the signal can be reconstituted completely using its decomposition [2], [3], [10]. Since wavelets carry orientation and multi-resolution information, it makes it an ideal tool for texture analysis and representation [4].

In major applications using subband decomposition, calculating each subband distribution constitutes a first step for designing efficient quantization and encoding algorithms [3], [8]. Recently, several works investigated the use of wavelets as a feature for texture retrieval. Essentially, these methods use either energy of the output filter banks [6] or wavelet subband distribution [1], [3], as a cue for texture discrimination. In the second approach, recent methods successfully investigated the use of the generalized Gaussian density (GGD) to represent the distribution of the wavelet coefficients. Compared to the Laplacien distribution or the Gaussian, the GGD has an additional free parameter that controls the shape of the distribution, which gives it more flexibility to fit different *platykurtic* or *leptokurtic* shapes of data. The authors in [1], [3] demonstrated the superiority of using the GGD representation over energy-based methods for texture retrieval. Furthermore, GGD representation has the additional advantage of combining feature extraction (FE) and similarity measurement (SM) in the same framework. That is, the texture retrieval problem is cast as a problem of finding texture images whose distribution models maximize the maximum likelihood (ML) (resp. minimize the KLD) with respect to the query data. However, the main assumption of these works is that a single GGD can capture the shape of the wavelet distribution in each subband decomposition. When examining several examples of texture images, one might clearly notice that for a wide range of natural texture images, wavelet distribution is overly heavy-tailed and the representation with a single GGD will lack accuracy.

For instance, when decomposing the image in Figure (1) into the first level Daubechies-4 orthogonal wavelets [2], we obtain the histogram of the coefficients shown in the same figure (solid line). After fitting the histogram with a single Gaussian, then a single GGD, respectively, we obtain the graphs shown in same figure (dashed lines). We can see clearly from the figure that the GGD fits better the shape of the histogram than the Gaussian. However, the histogram mode and tails still remain not well fitted by the GGD. In texture retrieval based on comparing wavelet distribution, for example [3], this may constitute a serious limitation since the missed information can play a critical role in texture discrimination.

Based on our previous work in [5], we propose in this paper to use a mixture of GGDs (denoted by MoGG) to represent the wavelet distribution for texture images. The MoGG has more flexibility than a single GGD to fit various shapes of data. Consequently, it allows to capture more complete texture information and,therefore, improve texture discrimination. We propose a new approach for calculating similarity between images using an approximation of the KLD using the Metropolis-Hastings sampling algorithm. Experimental results on a database of 1600 images have shown an average improvement of 20% in texture retrieval accuracy, compared to recent approaches, while maintaining a comparable level of computational complexity.



(a) (b)

Figure 1. Wavelets distribution approximation using: (a) a single Gaussian and (b) a single GGD; the solid line represents the coefficients histogram and the dashed line represents the histogram approximation.

This paper is organized as follows: Section II presents the MoGG representation of texture wavelets distribution. Section III presents our approach for texture retrieval using MoGG and KLD. Section IV presents some experimental results. We end the paper with a conclusion and future work perspectives.

II. Mixture of GGDs

The general Gaussian distribution for univariate random variable $\mathbf{x} \in \mathbb{R}$ is defined as follows:

$$p(\mathbf{x}|\mu, \sigma, \beta) = \frac{\beta\sqrt{\frac{\Gamma(3/\beta)}{\Gamma(1/\beta)}}}{2\sigma\Gamma\left(1/\beta\right)}\exp\left(-A(\beta)\left|\frac{\mathbf{x}-\mu}{\sigma}\right|^{\beta}\right) \quad (1)$$

where $A(\beta) = \left[\frac{\Gamma(3/\beta)}{\Gamma(1/\beta)}\right]^{\frac{\beta}{2}}$, $\Gamma(\cdot)$ denotes the gamma function, and μ and σ are the distribution mean and standard deviation parameters. The parameter $\beta \geq 1$ controls the tails of the pdf and determines whether it is peaked or flat: the larger the value of β, the flatter the pdf; and the smaller β is, the more peaked the pdf. This gives the pdf a flexibility to fit the shape of heavy-tailed data [5]. With a mixture of K GGDs, the marginal distribution of the random variable $\mathbf{x}$ is given by:

$$p(\mathbf{x}|\mathbf{\Theta}) = \sum_{i=1}^{K} \pi_i p(\mathbf{x}|\mu_i, \sigma_i, \beta_i) \quad (2)$$

where $0 < \pi_i \leq 1$ and $\sum_{i=1}^{K} \pi_i = 1$. To estimate the parameters $\mathbf{\Theta} = \{\pi_i, \mu_i, \sigma_i, \beta_i, i = 1, ..., K\}$, we use the Expectation-Maximization algorithm that we have proposed in [5]. Briefly, given a sample data $\mathcal{X} = \{\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_n\}$, the algorithm is designed to estimate the best model that fits $\mathcal{X}$ using the minimum message length principle. The optimal model provides the best tradeoff between data fitting accuracy and model complexity (i.e., the number of components K). The parameters of the model are given by the following iterative formulas:

$$\hat{\pi}_k = \frac{1}{n}\sum_{i=1}^{n} p(\theta_k|\mathbf{x}_i) \quad (3)$$

$$\hat{\mu}_k = \frac{\sum_{i=1}^{n} p(\theta_k|\mathbf{x}_i)|\mathbf{x}_i - \mu_k|^{\beta_k - 2}\mathbf{x}_i}{\sum_{i=1}^{n} p(\theta_k|\mathbf{x}_i)|\mathbf{x}_i - \mu_k|^{\beta_k - 2}} \quad (4)$$

$$\hat{\sigma}_k = \left[\frac{\beta_k A(\beta_k)\sum_{i=1}^{n} p(\theta_k|\mathbf{x}_i)|\mathbf{x}_i - \mu_k|^{\beta_k}}{\sum_{i=1}^{n} p(\theta_k|\mathbf{x}_i)}\right]^{1/\beta_k} \quad (5)$$

where θ_k denotes the parameters of the component k of the mixture and $A(\beta_k)$ is defined in Eq. (1). The posterior probability $p(\theta_k|\mathbf{x}_i)$ is given by: $p(\theta_k|\mathbf{x}_i) = \frac{\pi_k p(\mathbf{x}_i|\theta_k)}{\sum_{j=1}^{K}\pi_j p(\mathbf{x}_i|\theta_j)}$. Finally, the shape parameter β_k for each component is estimated simultaneously with the other parameters using the Newton-Raphson method:

$$\hat{\beta}_k \simeq \beta_k - \left\{\frac{\partial^2 \log(p(\mathcal{X}|\mathbf{\Theta}))}{\partial\beta_k^2}\right\}^{-1}\frac{\partial \log(p(\mathcal{X}|\mathbf{\Theta}))}{\partial\beta_k} \quad (6)$$

where $p(\mathcal{X}|\mathbf{\Theta})$ is the likelihood of generating the data by the model. Figure (2) shows the mixture representation of the same histogram in Figure (1). Clearly, we can see the improved accuracy in the histogram approximation.

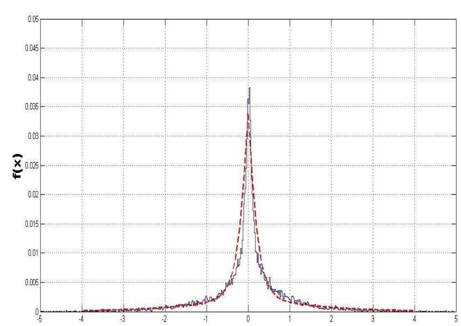


Figure 2. Approximation of the wavelet histogram using a MoGG.

III. Similarity measurement and texture retrieval

The problem of texture retrieval can be formulated as follows: Given a database containing $\mathcal{N}$ images and a query image $\mathcal{I}_q$, we aim to find the top $\mathcal{M}$ images ($\mathcal{M} \ll \mathcal{N}$) that are the most similar to the query image. Our solution for this part is inspired by the work in [3] which proposes to perform jointly feature extraction and similarity measurement for texture retrieval. Consider the query data which contain L points $\mathcal{X} = (\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_L)$ (the wavelet coefficients in our case), which are a sequence of i.i.d distributed data from the model $p(\mathcal{X}, \mathbf{\Theta}_q)$ of the query image. It has been established in [3], that the maximum likelihood selection is equivalent to minimizing the KLD between the distribution of the query and the target images, $p(\mathcal{X}, \mathbf{\Theta}_q)$ and $p(\mathcal{X}, \mathbf{\Theta}_i)$, which is given by:

$$D(p(\mathcal{X}, \mathbf{\Theta}_q), p(\mathcal{X}, \mathbf{\Theta}_i)) = \int p(\mathbf{x}, \mathbf{\Theta}_q) \log \frac{p(\mathbf{x}, \mathbf{\Theta}_q)}{p(\mathbf{x}, \mathbf{\Theta}_I)} d\mathbf{x} \qquad (7)$$

Note that a closed form of the above distance is not possible to obtain. Therefore, we resort to Monte Carlo sampling methods to calculate the integral of the KLD. The goal is to generate a sufficiently large sample $\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_n$ from the distribution $p(\mathbf{x}, \mathbf{\Theta}_q)$, in order to approximate the KDL distance using the Monte carlo integration:

$$D_{mc}(p(\mathcal{X}, \mathbf{\Theta}_q), p(\mathcal{X}, \mathbf{\Theta}_i)) = \frac{1}{n} \sum_{i=1}^{n} \log \frac{p(\mathbf{x}_i, \mathbf{\Theta}_q)}{p(\mathbf{x}_i, \mathbf{\Theta}_I)} \qquad (8)$$

which converges to $D(p(\mathcal{X}, \mathbf{\Theta}_q), p(\mathcal{X}, \mathbf{\Theta}_i))$ when $n \to \infty$. We propose to use Metropolis-Hastings algorithm [7] that generates a sample $\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_n$ according to a proposal distribution $Q(\mathbf{x}_i|\mathbf{x}_{i-1})$, which we suppose symmetric in our case. A candidate sample $\mathbf{x}_i$ is then accepted every time with the probability:

$$A(\mathbf{x}_i|\mathbf{x}_{i-1}) = min\left(1, \frac{p(\mathbf{x}_i, \mathbf{\Theta}_q)}{p(\mathbf{x}_{i-1}, \mathbf{\Theta}_q)}\right) \qquad (9)$$

Algorithm (1) summaries the main steps required in our approach for texture retrieval. Note that the sampling in step 2 is performed once for each subband in the query image. This step is not computationally expensive; practically, it takes few milliseconds to generate $n = 5 \times 10^4$ data points for all considered subbands.

Input : query image $\mathcal{I}_q$, database of $\mathcal{N}$ images.
Output: $\mathcal{M}$ top similar images ($\mathcal{M} \ll \mathcal{N}$).

1. Estimate the wavelet subband distribution $p(\mathbf{x}_i, \mathbf{\Theta}_q)$.
2. Sample n data points from the distribution $p(\mathbf{x}_i, \mathbf{\Theta}_q)$ using Eq. (9).
3. **for** $i \leftarrow 1$ **to** $\mathcal{N}$ **do**
4. Estimate the wavelet subband distribution $p(\mathbf{x}_i, \mathbf{\Theta}_i)$ for the image i;
5. Calculate the KLD using Eqs. (8) and (9).

end

6. Rank the images using the KLD so as to obtain the $\mathcal{M}$ top similar ones.

Algorithm 1: Wavelet-based texture retrieval.

IV. Experimental results

We conducted experiments on two texture databases, the MIT VisTex database [1], which contains 600 images grouped into 18 classes, and the database in [9] which contains 1000 images grouped into 25 classes. The images of both databases are of size 128×128 pixels (see Figure (3)). We used four Daubechies orthogonal wavelet [2], namely D_2, D_4, D_6 and D_8, and we applied three levels of decomposition for each texture image which gives nine independent subbands. The KLD between two images is therefore the sum of KLD of all the nine subbands. For the proposal distribution Q, we use a Gaussian with a scale $\sigma = 5$ determined experimentally. The algorithm is implemented entirely in a Matlab environment.

To measure the accuracy of texture retrieval, we compared our method with an energy-based approach and the method in [3] which uses a single GGD for texture representation. For the energy-based approach, we used features calculated using the $\mathbf{L}^1$ and $\mathbf{L}^2$ norms, respectively. Given wavelet coefficients $x_{i,1}$, $x_{i,1}$,..., $x_{i,L}$ in the *i*th subband, the following two values are used as features:

$$e_i^{(1)} = \frac{1}{L} \sum_{j=1}^{L} |x_{i,j}|, \quad e_i^{(2)} = \left(\frac{1}{L} \sum_{j=1}^{L} x_{i,j}^2\right)^{1/2} \qquad (10)$$

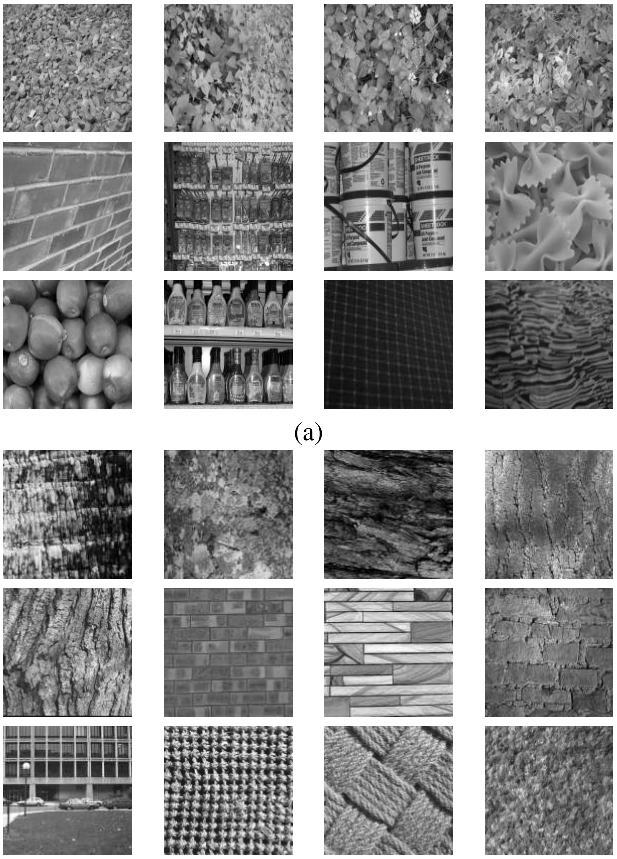
and the similarity measurement between the query and target images is calculated using the Euclidian distance between these feature values.

Table (I) shows comparative results for the three methods. The texture retrieval effectiveness is defined as the rate of retrieved relevant images to the actual number of relevant images in the database (which is named the *recall* measure). We give in the table the average value of 100 retrievals performed by randomly choosing a query image in the database. We can observe from the obtained results that using wavelet distributions comparison is always better than the energy-based approach for texture retrieval. However, our approach outperforms single GGD by almost 20% of accuracy. Finally, using three-levels instead of one-level decomposition enhances texture retrieval accuracy about 5%.

V. Conclusion

We presented a new statistical framework for texture representation and retrieval. Our approach, based on the MoGG model, improves significantly texture representation and discrimination accuracy. It compares favorably to recent methods using a single distribution parameters or wavelet-energy-based features. In the future, we intend to use the

[1]MIT Vision and modelling group: Vision texture, available at: http://vismod.media.mit.edu/vismod/imagery/VisionTexture/vistex.html



(a)

(b)

Figure 3. Sample images from: (a) VisTex and (b) the texture database in [9].

proposed approach for more general image and video representation and retrieval, as well as for texture synthesis.

ACKNOWLEDGMENT

The author would like to thank the Natural Sciences and Engineering Research Council of Canada for their support.

REFERENCES

[1] S-K. Choy and C-S. Tong. Supervised Texture Classification Using Characteristic Generalized Gaussian Density. *Journal of Mathematival Imaging and Vision*, 29(1):37-47, 2007.

[2] I. Daubechies. Orthonormal Bases of Compactly Supported Wavelets. *Communications on Pure and Applied Mathematics*, 41:909-996, 1988.

[3] M.N. Do and M. Vetterli. Wavelet-Based Texture Retrieval Using Generalized Gaussian Density and KL Distance. *IEEE Trans. on Image Processing*, 11(2):146-158, 2002.

[4] G. Fan and X.G Xia. Wavelet-Based Texture Analysis and Synthesis Using Hidden Markov Models. *IEEE Trans. on Circuits and Systems-I*, 50(1):106-120, 2003.

Filters	Methods		
	$\mathbf{L}^1 + \mathbf{L}^2$	GGD&KLD	MoGG&KLD
D_2	55.06	69.20	**85.63**
D_4	54.02	66.62	**84.65**
D_6	53.89	64.89	**84.36**
D_8	53.25	63.94	**82.62**

(a) One-level decomposition.

Filters	Methods		
	$\mathbf{L}^1 + \mathbf{L}^2$	GGD&KLD	MoGG&KLD
D_2	60.52	70.12	**89.62**
D_4	59.20	70.63	**87.25**
D_6	59.32	69.39	**88.22**
D_8	59.12	69.14	**87.12**

(b) Two-level decomposition.

Filters	Methods		
	$\mathbf{L}^1 + \mathbf{L}^2$	GGD&KLD	MoGG&KLD
D_2	61.30	71.56	**90.98**
D_4	60.09	70.29	**89.85**
D_6	59.33	68.97	**89.65**
D_8	59.36	68.42	**88.23**

(c) Three-level decomposition.

Table I

AVERAGE RETRIEVAL ACCURACY IN THE TOP 20 IMAGES FOR DIFFERENT FILTERS AND DECOMPOSITION LEVELS.

[5] M.S. Allili, N. Bouguila and D. Ziou. Finite General Gaussian Mixture Modelling and Application to Image and Video Foreground Segmentation. *Journal of Electronic Imaging*, 17:013005, 2008.

[6] T. Randen and J-H. Husoy. Filtering for Texture Classification: A Comparative Study. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 21(4):291-310, 1999.

[7] C.P. Robert and G. Casella. *Monte Carlo Statistical Methods*, Springer, 2004.

[8] M. Vetterli and J. Kovacevic. *Wavelets and Subband Coding*. Prentice hall, 1995.

[9] Y. Xu and J. Hui and C. Fermuller. A Projective Invariant for Textures. *IEEE Conf. Computer Vison and Pattern Recognition*, 1932-1939, 2006.

[10] G. Van De Wouwer and P. Scheunders and D. Van Dyck. Statistical Texture Characterization From Discrete Wavelet Representation. *IEEE Trans. on Image Processing*, 8(4):592-598, 1999.