Texture Similarity Using Periodically Extended and Adaptive Curvelets

Hasan Al-Marzouqi and Ghassan AlRegib,
School of Electrical and Computer Engineering,
Georgia Institute of Technology,
Atlanta, GA 30332, USA.

***Abstract*—This paper presents a new method for texture based image retrieval. The proposed algorithm uses a periodically extended variant of the curvelet transform. The sum of the absolute value of differences in the mean and standard deviation between curvelet wedges representing the query image and the test image is used as the distance index. Performance improvement is demonstrated using the CUReT database, where the proposed algorithm significantly outperforms previously proposed methods that were based on Curvelet, Gabor, LBP, and wavelet features. We also show that adapting curvelet tiles increases the performance of the proposed method.**

***Index Terms*—CBIR, Texture, Curvelet**

I. Introduction

Efficient techniques for searching, browsing and retrieval of image and video data sets can make significant changes to the way we access such material today. The use of perceptual and visual properties can highly improve the efficiency of browsing and indexing such visual content. Textures are important elements in the perception of a variety of visual information. Several methods to assess the similarity of textures were proposed in the literature. Do and Vetterli [1] use the Kullback-Leibler distance between generalized Gaussian density (GGD) estimates of wavelet subbands. Alternatively, The energy and gradient of wavelet coefficients are used by Huang and Dai [2]. Manjunath and Ma [3] proposed a texture retrieval method based on the l_1 norm between Gabor features representing query and test images. The extracted feature vectors are composed from the mean and variance of Gabor coefficients. Zhang *et. al.* [4] report improvements over Gabor filters by using curvelet features. The default variant of curvelets used in Zhang *et. al.* work doesn't use periodic extension or angular divisions in the outer curvelet scale. Zujovic *et. al.* [5] use statistical properties of a steerable pyramid representation of the texture data set. The statistical properties used include: the mean, variance, horizontal and vertical autocorrelation, and crossband correlations. Ojala *et. al.* [6] resort to spatial domain methods, where they capture rotation invariant statistical features based on local binary patterns.

In this paper, we propose new texture retrieval algorithms using a periodically extended variant of the curvelet transform. Similar to Gabor wavelets and steerable pyramids, the curvelet transform is a multi-scale directional transform that divides the spatial content of the signal of interest into different scales and directional sub-bands. Curvelets provide an efficient representation of edges and other directional features in images.

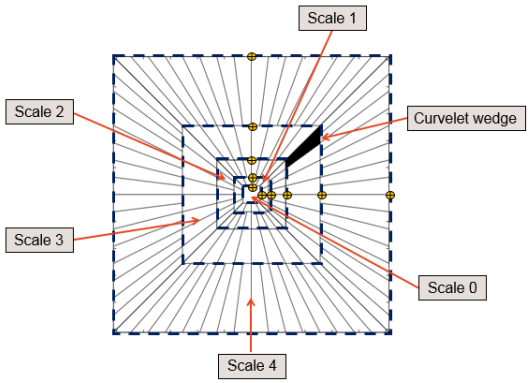

Fig. 1. Default curvelet tiling. 5-scale decomposition is shown. Each scale is divided into $16 \times 2^{\lceil (j-1)/2 \rceil}$ wedges, where j is the scale number. Scale locations are denoted by yellow markers.

Periodic extension reduces boundary artifacts by connecting border elements. We propose three different algorithms that use the periodically-extended curvelet transform for texture retrieval. The first algorithm uses the default dyadic construction of curvelet subbands. It is also faithful to the original angular divisions proposed by curvelet authors. The remaining algorithms use adaptive curvelets. Adapting curvelets [7]–[9] improves the default curvelet representation by making the frequency bands and directions of curvelet wedges dependent on image content. The feature vector we propose using for image retrieval is constructed from the mean and standard deviation of curvelet wedges.

The remainder of the paper is organized as follows. In the next sections, we will present an overview of default and adaptive curvelet transforms. This will be followed by the technical details of our texture retrieval algorithm in Section IV. Experimental results are presented in Section V. Finally, Section VI presents our conclusions.

II. Default Curvelets with Periodic Extension

The curvelet transform [10] provides an efficient representation of directional features. It works by dividing the spatial content of images into different frequency bands representing unique scale and directional features. The curvelet transform has shown successful performance improvements in a wide range of application areas including denoising [8], [11], [12],

978-1-4799-7088-9/14/$31.00 ©2014 IEEE

image fusion [13], edge enhancement [14], compressed sensing data recovery [9], [15], deconvolution [16], texture analysis [17] and face recognition [18], [19].

Curvelet coefficients are generated using the 2D fast Fourier transform (FFT). FFT is first applied to an image. Next, the 2D Fourier Transform (FT) plane is divided into different scales. The number of scales depends on the size of the image and is given by

$$J = \lceil \log_2(\min(N_1, N_2) - 3) \rceil, \quad (1)$$

where $\lceil * \rceil$ is the ceiling function, and N_1 and N_2 are the number of vertical and horizontal image pixels respectively. The subtraction of three in (1) ensures a minimum dimension of 8×8 in the most inner scale. Scale locations, which we define as coordinates of the FT plane that determine the outer boundary of each scale, are determined in a dyadic manner. Each curvelet scale is further divided into a number of different directional wedges as shown in Figure 1. Curvelet coefficients are generated by taking the IFFT's of these wedges.

Since Fourier transform values for real data are symmetric around the center of the FT plane (*D.C.* value), two quadrants are sufficient for constructing the curvelet representation. To obtain real-valued coefficients for such data sets, the complex coefficients are separated into two parts making the total number of wedges unchanged.

Periodic extension is used in the outer level to reduce boundary artifacts. An alternate implementation of default curvelets replace the angular divisions at the outer scale by a high-pass filter. This implementation reduces computational cost but loses some directional information. Curvelets constructed with outer angular divisions and periodic extension are used in the proposed texture retrieval algorithms.

III. Adaptive Curvelets

Adaptive curvelets improve over default curvelets by adapting the frequency divisions to better represent the content of a given image. In adaptive curvelets, the frequency tiling that provides maximal improvements in cost function values is considered optimal [8]. Adaptation parameters include the number of angular divisions per scale/quadrant pair and scale locations. The optimal choice for these parameters is found using an optimization algorithm maximizing a cost function. Two cost functions are considered in this work. First, denoising performance as measured by (PSNR). Using this approach the query image is contaminated with Additive White Gaussian Noise (AWGN) with standard deviation σ that is given be the following equation:

$$\sigma = \max(\textit{std(I)}, 0.05 \times \textit{MAXI}), \quad (2)$$

where $std(I)$ is the standard deviation of image I intensities, and *MAXI* is the maximum possible intensity value. The lower bound for σ values used in (2) is found to improve the performance and robustness of the adaptation algorithm. The curvelet tiling that generates the best denoising performance is considered optimal as it indicates a better representation of image features. The second cost function considered is the coefficient of variation C_v of curvelet coefficients magnitude. The coefficient of variation is defined as the ratio between the standard deviation of a vector's elements and its mean $C_v = \sigma/\mu$. Maximizing the coefficient of variation increases the sparsity of curvelet coefficients. Note that since curvelet tiling adaptations alter the length of the coefficients vector, minimizing l^p norms will not generate satisfactory results

Angular decompositions are optimized separately from scale location. A multi-resolution global optimization algorithm combining angular optimization and scale locations search is used to find the optimal adaptive curvelet parameters.

1) Optimal Number of Angular Divisions per Scale/Quadrant: Optimizing the number of angular divisions is performed using an exhaustive search procedure for each scale/quadrant pair. The tested parameters are chosen from the following sequence {4, 8, 12, 16, 20, 24}. The divisions are uniformly distributed in each scale/quadrant pair. Recalling FFT's symmetry for real data the number of parameters to optimize is $2\times$*(J-1)* for real data, and is equal to $4\times$*(J-1)* for the case of complex input data. The distribution of angular decompositions that generates the maximum PSNR or C_v value is considered optimal.

2) Optimal Scale Locations: Derivative-free optimization methods are used to find the optimal scale locations. Such methods start at an initial point x_0, evaluate the cost function at a selected mesh of points in the neighborhood of x_0, and define the point in the mesh with the minimum function value as the new x_0. Next, the algorithm iterates until a specified convergence criteria is met. The Nelder-Mead simplex search method [20], [21] is one of the popular methods for generating such a mesh. It has been used extensively in a variety of application areas. It is used in this work to find the optimal scale locations. Given a specific angular distribution the scale locations search algorithm finds optimal vectors **H** and **V**, where $\mathbf{V} = \{V_1, V_2, ..., V_J\}$ are the coordinates of vertical scale locations, and $\mathbf{H} = \{H_1, H_2, ..., H_J\}$ are the horizontal scale locations

3) Global Optimization: In this part, we introduce the global optimization algorithm that combines the two previous adaptations. The algorithm uses a multi-resolution search strategy. This helps in avoiding convergence to local minima, and reduces the computational cost of the algorithm. Optimization is done in a hierarchical manner consisting of n iterations. In each iteration, optimal scale and angular locations are found for a downsampled smoothed version of the input image. The flow of this algorithm is illustrated in Figure 2.

The algorithm starts with downsampling the original image $n-1$ times by a factor of two. The image is also smoothed by a Gaussian filter. This smoothing operation is performed to reduce possibilities of convergence to local minima. The optimal angular decomposition is found using dyadic scale locations and the optimal number of scales computed using (1). The angular decompositions found and the dyadic scale locations are used as initial points for the scale locations search algorithm. Next, the value of n is decremented by one and a new loop starts. Initial values for scale locations

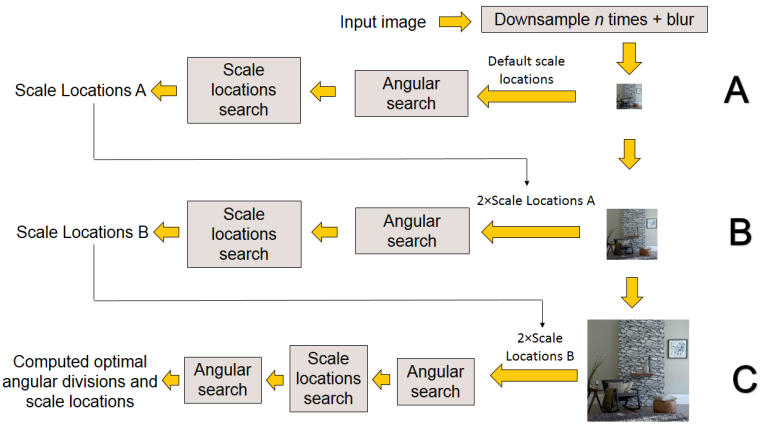

Fig. 2. Flow diagram of the global optimization algorithm.

will be the scale locations computed in the previous iteration after being re-scaled to the new image coordinates. After n iterations, once the scale locations search algorithm converges to a minimum, the angular decomposition algorithm is run to search for the optimal angular decomposition at the calculated scale locations. Finally, the algorithm terminates returning the determined optimal scale and angular decompositions. The returned tiles are going to be used in the next section to extract feature vectors for use in texture retrieval applications.

IV. Feature Vectors and Distance Computation

Forming a representative feature vector is essential for the success of transform domain retrieval methods. Using the curvelet transform, the query and input images are represented by sets of coefficients identifying the spatial content relative to unique frequency bands. A curvelet element is represented in the spatial domain by a needle-shaped elongated object as can be seen in Figure 3. The figure was created using a four decomposition level default inverse curvelet transform of a set of zero valued curvelet coefficients except for the center of the first tile (out of 32 tiles) at the third curvelet scale. The coefficient magnitude at the center of the aforementioned tile was set to one. Such elongated objects describe the edge content of images. The scale and the direction of the curvelet tiles specify the thickness and the direction of curvelet elements. The magnitudes of curvelet coefficients scale the intensity of curvelet elements.

The similarity between two curvelet wedges representing a certain frequency band is indicative of the similarity of the spatial content in this band between the two images. Therefore, a global similarity measure between two images can be formed by including the similarities between curvelet wedges across all frequency bands. Given the fact that edges are essential components in images, a curvelet-based similarity measure between curvelet wedges is perceptually meaningful. Furthermore, adapting curvelet coefficients to better represent the query image constructs a representation from curvelet objects that are more faithful in direction and scale to the features of the image of interest.

Fig. 3. A curvelet element in the spatial domain

Given two images A and B, where A is the query image, we propose three algorithms for texture retrieval. The first algorithm uses default curvelet with periodic extension and angular divisors in the outer curvelet scale. The remaining algorithms use adaptive curvelets. In adaptive curvelets, the optimal curvelet tiling representing a query image A is computed using one of the cost functions mentioned in the previous section (maximizing denoising performance or C_v). Next, the computed optimal tiling is used to compute the curvelet coefficients representing images A and B.

The feature vectors describing images A and B is formed by the mean and the standard deviation of curvelet coefficients in each wedge. Let the number of curvelet wedges in the optimal representation be K. The feature vector representing images A and B is given by

$$F = \{\mu_1, \sigma_1, \mu_2, \sigma_2, ..., \mu_K, \sigma_K\}, \quad (3)$$

where μ_i is the mean of curvelet coefficient in wedge i and σ_i is the standard deviation of the coefficients in the same wedge.

Next, the distance between images A and B is computed by

$$D(A, B) = \|F(A) - F(B)\|_1, \quad (4)$$

where $\|x\|_1$ is the l_1 norm of vector x. Before finishing this section, we would like to note that the feature vectors can be weighted to (de-)emphasize certain scales and orientation of interest (*e.g.* coarse/fine or horizontal/vertical features). This

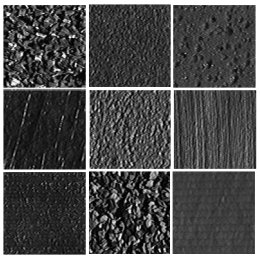

Fig. 4. Sample images from the CUReT texture database

TABLE I
PERFORMANCE COMPARISON OF THE DIFFERENT ALGORITHMS USED IN THIS STUDY.

	P@1	P@2	MRR	MAP
Gabor features	0.934	0.869	0.962	0.901
Wavelet features	0.967	0.861	0.980	0.924
Default curvelet	0.967	0.861	0.983	0.918
Linear Binary Patterns (LBP)	0.934	0.885	0.959	0.925
Curvelet with periodic extension	**1.000**	0.934	**1.000**	0.964
Denoising based adaptive curvelets	**1.000**	0.951	**1.000**	0.969
C_v based adaptive curvelets	**1.000**	**0.959**	**1.000**	**0.975**

path is not pursued in this paper and equation (3) is the one used in forming feature vectors.

V. RETRIEVAL RESULTS

A. Experimental setup

The performance of the proposed algorithms is tested using the Columbia-Utrecht Reflectance and Texture Database (CUReT) [22]. The database contains 61 images of real-world surfaces taken at different illumination and viewing directions. In our experiments we used images taken at the first illumination setting with viewing direction 22. CUReT images are of size 640 $\times$ 480. They all include a textural part and a background. Three images of size 128 $\times$ 128 covering the textural regions were extracted from each CUReT image. One of these images is selected randomly as a query image. The remaining images are used as testing images. A sample of the images used in this study is shown in figure 4

B. Retrieval performance

The performance of the three proposed methods is compared with the following algorithms:

1) l^2 norm of default curvelet features [4]. This algorithm uses no periodic extension or angular divisions in the outer curvelet scale
2) l^1 norm of Gabor features [3]
3) Kullback-Leibler distance on wavelet features [1]
4) Log-likelihood ratio of Local Binary Patterns [6]. The used feature vector is composed from the sum of $LBP^{riu2}_{8,1}$ and $LBP^{riu2}_{24,3}$ features.

Table I reports the performance of the tested algorithm using three different retrieval measures. Precision at one (P@1) is the average ratio of the number of matches where the first retrieved image is relevant to the total number of correct matches. Similarly, precision at two (P@2) is the average precision of

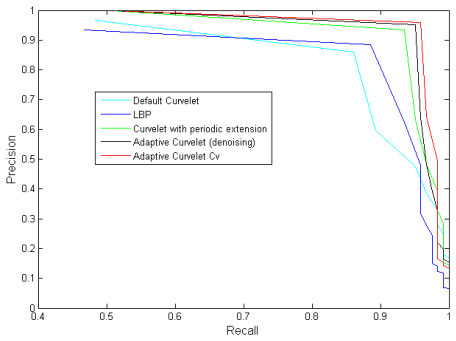

Fig. 5. Precision-recall curves for five different algorithms used in this study

the second retrieved image. Mean reciprocal rank (MRR) is the mean reciprocal of the rank of the first relevant position. Mean Average Precision (MAP) is the mean of average precision values across all queries. The average precision (AP) is given by

$$AP = \frac{\sum_{\forall k} P(k)}{N}, \tag{5}$$

where k and N are the ranks and number of relevant images, respectively.

Precision at one results indicate that the proposed algorithms succeed in retrieving one of the correct matches as a first retrieved image. Precision at two (P@2) results show that the proposed methods are able to retrieve 93-96% of the correct matches in our database. The proposed methods outperform the other algorithms in MAP by a difference of at least 4%. Precision-recall curves for methods based on LBP, default curvelet, curvelet with periodic extension, denoising and C_v based curvelets are shown in figure 5.

VI. CONCLUSIONS

Using curvelets we were able to achieve results outperforming well known texture retrieval algorithms. The improved result show that curvelet divisions of image content are well suited for retrieval applications. The use of outer extension window further improve curvelet's performance by eliminating boundary artifacts. Our results strongly encourage the use of periodically extended curvelets despite the increase in the number of coefficients and computational cost. Adapting curvelet tiles succeed in increasing the performance of default curvelets. To reduce computational costs, training of adaptive curvelets can be performed prior to receiving image queries. In this case, the search for optimal curvelet tiles is going to be performed over a set of images representing the class of interest. The texture database will consist of the images representing various material classes along with the optimal curvelet representation found per class. The robustness of mean and standard deviation based feature vectors can be improved by normalizing their elements by the mean and the standard deviation of curvelet coefficients in the entire database. In the future, we plan to extend the use of the developed methods to larger image and video data sets.

REFERENCES

[1] M. Do and M. Vetterli, "Texture similarity measurement using kullback-leibler distance on wavelet subbands," in *Image Processing, 2000. Proceedings. 2000 International Conference on*, vol. 3, 2000, pp. 730–733 vol.3.

[2] P. Huang and S. Dai, "Design of a two-stage content-based image retrieval system using texture similarity," *Information Processing and Management*, vol. 40, no. 1, pp. 81 – 96, 2004. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0306457302000973

[3] B. Manjunath and W. Ma, "Texture features for browsing and retrieval of image data," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 18, no. 8, pp. 837–842, Aug 1996.

[4] D. Zhang, M. Islam, G. Lu, and I. Sumana, "Rotation invariant curvelet features for region based image retrieval," *International Journal of Computer Vision*, vol. 98, no. 2, pp. 187–201, 2012. [Online]. Available: http://dx.doi.org/10.1007/s11263-011-0503-6

[5] J. Zujovic, T. Pappas, and D. Neuhoff, "Structural texture similarity metrics for image analysis and retrieval," *Image Processing, IEEE Transactions on*, vol. 22, no. 7, pp. 2545–2558, July 2013.

[6] T. Ojala, M. Pietikainen, and T. Maenpaa, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 24, no. 7, pp. 971–987, Jul 2002.

[7] H. Al-Marzouqi and G. AlRegib, "Curvelet transform with adaptive tiling," in *Proc. SPIE*, vol. 8295, 2012. [Online]. Available: http://dx.doi.org/10.1117/12.909111

[8] ——, "Searching for the optimal curvelet tiling," in *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*, May 2013, pp. 1626–1630.

[9] ——, "Using the coefficient of variation to improve the sparsity of seismic data," in *Global Conference on Signal and Information Processing (GlobalSIP), 2013 IEEE*, Dec 2013.

[10] E. Candes, L. Demanet, D. Donoho, and L. Ying, "Fast discrete curvelet transforms," *Multiscale Modeling & Simulation*, vol. 5, no. 3, pp. 861–899, 2006.

[11] J. Starck, E. Candès, and D. Donoho, "The curvelet transform for image denoising," *Image Processing, IEEE Transactions on*, vol. 11, no. 6, pp. 670–684, 2002.

[12] B. Saevarsson, J. Sveinsson, and J. Benediktsson, "Combined wavelet and curvelet denoising of SAR images," in *Geoscience and Remote Sensing Symposium, 2004. IGARSS'04. Proceedings. 2004 IEEE International*, vol. 6. IEEE, 2004, pp. 4235–4238.

[13] M. Choi, R. Kim, M. Nam, and H. Kim, "Fusion of multispectral and panchromatic satellite images using the curvelet transform," *Geoscience and remote sensing letters, IEEE*, vol. 2, no. 2, pp. 136–140, 2005.

[14] M. Miri and A. Mahloojifar, "Retinal image analysis using curvelet transform and multistructure elements morphology by reconstruction," *Biomedical Engineering, IEEE Transactions on*, vol. 58, no. 5, pp. 1183–1192, 2011.

[15] F. Herrmann and G. Hennenfent, "Non-parametric seismic data recovery with curvelet frames," *Geophysical Journal International*, vol. 173, no. 1, pp. 233–248, 2008.

[16] M. Fadili and J.-L. Starck, "Sparse representation-based image deconvolution by iterative thresholding," *Astronomical Data Analysis ADA*, vol. 6, p. 18, 2006.

[17] S. Arivazhagan, L. Ganesan, and T. S. Kumar, "Texture classification using curvelet statistical and co-occurrence features," in *Pattern Recognition, 2006. ICPR 2006. 18th International Conference on*, vol. 2. IEEE, 2006, pp. 938–941.

[18] T. Mandal, Q. Jonathan Wu, and Y. Yuan, "Curvelet based face recognition via dimension reduction," *Signal Processing*, vol. 89, no. 12, pp. 2345–2353, 2009.

[19] Zhihua-Xie, G. Liu, S. Wu, and Y. Lu, "A fast infrared face recognition system using curvelet transformation," in *Electronic Commerce and Security, 2009. ISECS '09. Second International Symposium on*, vol. 2, 2009, pp. 145–149.

[20] A. A. R. Conn, K. Scheinberg, and L. N. Vicente, *Introduction to derivative-free optimization*. SIAM, 2009, vol. 8.

[21] J. A. Nelder and R. Mead, "A simplex method for function minimization," *The computer journal*, vol. 7, no. 4, pp. 308–313, 1965.

[22] K. Dana, B. Van-Ginneken, S. Nayar, and J. Koenderink, "Reflectance and Texture of Real World Surfaces," *ACM Transactions on Graphics (TOG)*, vol. 18, no. 1, pp. 1–34, Jan 1999.