Steerable Pyramid Decomposition – Rotation & Scale invariant texture image retrieval

G.Pandiselvi, V.Umamaheswari, S.Pethammal Jothi, C.Balasubramanian

P.S.R.Rengasamy College Of Engineering For Women, Sivakasi

Abstract

A new rotation- invariant and scale invariant representation for texture image retrieval process on steerable pyramid decomposition. To obtain rotation or scale invariance, the feature elements are aligned by considering either the dominant orientation or dominant scale of the input textures. Initially, take a various train images (data samples) then extract the various features from that rotational texture images and stored in data base. Similarly test the images, then extract the features of text images and compare with data base based similarity features we can extract image (similar) from the data base. In test Experiments were conducted on the broad database aiming to compare our approach to the conventional steerable pyramid decomposition, and a recent proposal for texture characterization based on Gabor wavelets with regard to their retrieval effectiveness. Results demonstrate the maximum similarity images are extracted from the data base and conclude the image retrieval application using feature extraction basis.

Index Terms

Texture classification, feature extraction, steerability, rotation invariance, Gabor wavelet.

1. Introduction

Texture can be broadly defined as the visual or tactile surface characteristics and appearance of something. Texture is an important characteristic for analysis of many types of images. Texture is present in many real as well as artificial data. Its importance and present everywhere in image data a formal approach or definition of texture analysis does not exist. Texture is a natural property of almost all surfaces the grain of wood, the pattern of crop in fields etc. It contains important information about the structural arrangement of surfaces and their relationship to the surrounding environment.

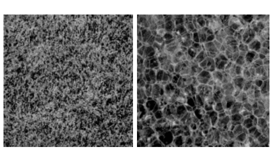

Fig 1: Sample Texture images.

Texture classification tasks involve two main steps: (1) Feature extraction step, where texture features are extracted from the image and (2) Classification step, where texture class membership is assigned according to the extracted texture features. A feature is a characteristic that can capture a certain visual property of an image either globally for the whole image, or locally for objects or regions. texture image retrieval applications consists therefore in achieving rotation- and scale- invariant feature representations for non-controlled environments. Texture may be coarse, fine, and smooth, granulated, rippled regular, irregular, or linear

One of the approaches to texture feature extraction is the filter bank approach that decomposes a texture image into subbands using a linear transform or filter bank. Several previous works extract texture features based on wavelet packet signatures and wavelet frames. Although these methods allow for a multiresolution decomposition, they are limited in directional selectivity and not able to capture directional information. Two patterns that are visually very different can have identical global distributions of scales and directions. Two patterns that are visually very different can have identical global distributions of scales and directions. The multiresolution theory of the wavelet transform provides an elegant solution to the locality problem for scale characterization.

Many existing systems do not care about such variations or they handle it in a very limited way.In texture analysis, rotation and illumination invariance plays a great attention. Many researchers have been done on rotation and illumination invariance. There are various algorithms, such as GLCM [1], Gabor filters [2], wavelet transforms [3], Markov random field [4], Many algorithms for texture classification are not rotation and illumination invariant. The effectiveness of a texture classification algorithm can be increased by using a module for feature extraction followed by classification. Classification we are using two different methods one is Support Vector Machine and another is k- nearest neighbor and we are find that by using which classifier it gives better result.

Manuscript received March 5, 2016
Manuscript revised March 20, 2016

2. Related work

Image retrieval is very interesting and vast field. Since 1970, research on the advance image retrieval is started. In 1979, a conference on Database Techniques for Pictorial Applications was held in Florence due to which many researchers attracted towards the field of image database management. After that several researches had been done on features based image retrieval, later a system [7] was proposed which uses the concept of texture based image retrieval system combines with the wavelet decomposition [12] and gradient vector [18].

In that system every image is associated with a coarse feature descriptor and a fine feature descriptor was derived through the use of wavelet coefficients related to the original image. In the first stage coarse feature descriptor is used so that non-promising images can be quickly separated, so that searching for similar images can be done efficiently. Another image retrieval system was introduced [4] which was based on the principle of motif co-occurrence matrix (MCM), which can easily find out the basic difference between pixels and can also convert them into a basic graphic. It can compute the probability of occurrence of pixels in the adjacent area and work as an image feature.

Another system was proposed which utilizes the properties like contrast [14], coarseness and directionality models [17, 13] to achieve texture classification and recognition. After that, a texture-based image retrieval method was proposed which was based on two-stage content-based image retrieval system by using texture similarity[5] which enhanced the image retrieval technique. To obtain better result, we have proposed a better retrieval technique which integrated color and texture features in order to improve image retrieval.

Color histogram [19] is one of the common techniques used in image retrieval systems texture features are extracted by using the concept of Pyramid Structure Wavelet Transformation combined with Euclidean Distance from those images which was previously classified through color model. The result obtained by using this system is better than other convention systems which only use color, texture features individually. Hence we can say that combination of color and texture feature for finding similar image retrieval makes system more efficient and effective. . Manual image annotation is time-consuming, laborious and expensive; to address this, there has been a large amount of research done on automatic image annotation. Additionally, the increase in social web applications and the semantic web have inspired the development of several web-based image annotation tools.

3. Proposed System

1. Input images:

Here the input images are texture images. The term texture generally refers to repetition of basic texture elements called Texel's. The Texel contains several pixels, whose placement could be periodic, quasi-periodic or random.

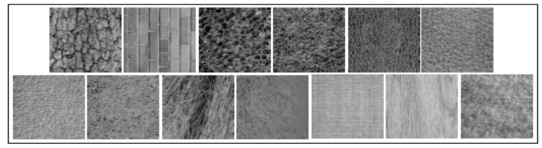

Figure 2 Texture images

Texture may be coarse, fine, and smooth, granulated, rippled regular, irregular, or linear. From left to right, and from top to bottom, they include: Bark, Brick, Bubbles, Grass, Leather, Pigskin, Raffia, Sand, Straw, Water, Weave, Wood, and Wool From this database, three different image datasets were generated: non-distorted, rotated-set A, and rotated-set B. The non-distorted image dataset was constructed just from the original input textures. Each texture image was partitioned into sixteen 128×128 non-overlapping sub images. The second image dataset is referred to as rotated image dataset A, and was generated by selecting the four 128×128 innermost sub images from texture images at 0, 30, 60, and 120 degrees.

2. Steerable pyramid decomposition:

The steerable pyramid decomposition is a linear multi resolution image decomposition method, by which an image is subdivided into a collection of sub bands localized at different scales and orientations. Using a high- and low- pass filter (H0, L0) the input image is initially decomposed into two sub bands: a high- and low-pass sub bands, respectively. Further, the low pass sub band is decomposed into K-oriented band-pass portions B0…....Bk-1, and into a low-pass sub band L1. The decomposition is done recursively by sub sampling the Lower low-pass sub band (Ls) by a factor of 2 along the rows and columns. Each recursive step captures different directional information at a given scale.

The basis functions of the steerable pyramid are directional derivative operators, that come in different sizes and orientations. An example decomposition of an image of a white disk on a black background is shown to the right. This particular steerable pyramid contains 4 orientation sub bands, at 2 scales. The number of orientations may be adjusted

by changing the derivative order.

The smallest sub band is the residual low pass information. The residual high pass sub band is not shown. The block diagram for the decomposition (both analysis and synthesis) is shown to the right. Initially, the image is separated into low and high pass sub bands, using filters L0 and H0.

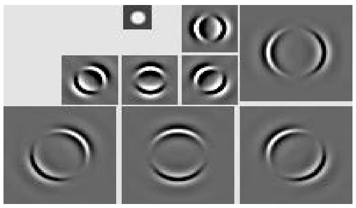

Figure 3 Decomposition in texture images

The dominant orientation (DO) is defined as the orientation with the highest total energy across the different scales considered during image decomposition. It is decomposed by finding the highest accumulated energy for the K different orientations considered during image decomposition

$$DO_i = \max\left\{E_0^{(R)}, E_1^{(R)}, \ldots, E_{(K-1)}^{(R)}\right\}$$

Where I is the index where the dominant orientation appeared, and:

$E_n^{(R)} = \sum_{M=0}^{S-1} E(m,n)$, n=0,1……k-1

Note that each En(R) covers a set of filtered images at different scales but at same orientation. The low pass sub band is then divided into a set of oriented band pass sub bands and a low(er)-pass sub band. This low(er)-pass sub band is sub sampled by a factor of 2 in the X and Y directions.

The recursive (pyramid) construction of a pyramid is achieved by inserting a copy of the shaded portion of the diagram at the location of the solid circle (i.e., the low pass branch). The steerable pyramid performs a polar-separable decomposition in the frequency domain, thus allowing independent representation of scale and orientation.

4. Feature extraction:

i. Histogram:

In a more general mathematical sense, a histogram is a mapping mi that counts the number of observations that fall into various disjoint categories (known as sbins), whereas the graph of a histogram is nerely one way to represent a histogram. Thus, if we let n be the total number of observations and k be the total number of bins, the histogram mi meets the following conditions.

ii. Standard deviation:

There are two common textbook definitions for the standard deviation s of a data vector X.

$$s = \left[\frac{1}{n-1}\sum_{i=1}^{n}(x_i - \bar{x})^2\right]$$

Where

$$\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i$$

The two forms of the equation differ only in versus in the divisor. s = std(X), where X is a vector, returns the standard deviation. The result s is the square root of an unbiased estimator of the variance of the population from which X is drawn, as long as X consists of independent, identically distributed samples. If X is a matrix, std(X) returns a row vector containing the standard deviation of the elements of each column of X. If X is a multidimensional array, std(X) is the standard deviation of the elements along the first non singleton dimension of X.

s = std(X, flag) for flag = 0, is the same as std(X). For flag = 1, std(X, 1) returns the standard deviation using (2) above, producing the second moment of the set of values about their mean. s = std(X, flag, dim) computes the standard deviations along the dimension of X specified by scalar dim. Set flag to 0 to normalize Y by n-1; set flag to 1 to normalize by n

iii. Mean:

Average or mean value of array is known as mean.

Syntax:

M = mean (A)

M = mean (A, dim)

M = mean (A) returns the mean values of the elements along different dimensions of an array. If A is a vector, mean (A) returns the mean value of A. If A is a matrix, mean (A) treats the columns of A as vectors, returning a row vector of mean values.

If A is a multidimensional array, mean (A) treats the values along the first non-singleton dimension as vectors, returning an array of mean values. M = mean (A, dim) returns the mean values for elements along the dimension of A specified by scalar dim. For matrices, mean (A, 2) is a column vector containing the mean value of each row.

Example: If X = [0 1 2 3 4 5]

Then mean(X, 1) is [1.5 2.5 3.5] and

Mean(X, 2) is [1 4]

5. Classification

Texture classification refers to the process of grouping test samples of texture into classes, where each resulting class contains related samples according to some similarity criterion. The goal of classification in general is to select the most appropriate category for an unknown object, given a set of known categories. While perfect classification is frequently impossible, the classification may also be performed by determining the probability for each of the known categories. There are three major groups of classifiers are popularly used, including k-

Nearest Neighbors, Artificial Neural Networks (ANN) and Support Vector Machines (SVM). But in our work we used k-NN and SVM classification methods. So, here we introduced some concept of that svm classifiers.

A. *SVM*

The original SVM algorithm was invented by Vladimir N. Vapnik and the current standard soft margin was proposed by Vapnik and Corinna Cortes in 1995. SVM is supervised learning classifier. SVM are the newer trends in machine learning algorithm which is popular in many pattern recognition problems in current years, as well as texture classification. SVM is designed to maximize the marginal distance between classes with decision boundaries drawn using different kernels. SVM is designed to work with only two classes by determining the hyper plane to divide two classes. This is prepared by maximizing the margin from the hyper plane to the two classes. The samples nearest to the margin that were selected to determine the hyper plane is known as support vectors. Multiclass classification is also applicable, the multiclass SVM is basically built up by various two class SVMs to solve the problem, either by using one versus all or one versus one. The winning class is then determined by the highest output function or the maximum votes

respectively. Despite that, SVM is still considered to be powerful classifier which was replacing the ANN and has slowly evolved into one of the most important main stream classifier. They are now widely used in the research of texture classification.

6. Similarity measure:

Similarity between images is obtained by computing the distance of their corresponding feature vectors. The smaller the distance, the more similar the images. Given the query image (i), and the target image (j) in the dataset, the distance between the two patterns

$$\mathrm{d}(\mathrm{i},\mathrm{j}) = \sum_{\mathrm{m}} \sum_{\mathrm{m}} \mathrm{d}_{\mathrm{mn}}(\mathrm{i},\mathrm{j})$$

Where:

$$\mathrm{d}_{\mathrm{mn}}(\mathrm{i},\mathrm{j}) = \left|\frac{\mu_{\mathrm{mn}}^{2} - \mu_{\mathrm{mn}}^{3}}{\alpha(\mu_{\mathrm{mn}})}\right| + \left|\frac{\sigma_{\mathrm{mn}}^{2} - \sigma_{\mathrm{mn}}^{3}}{\alpha(\sigma_{\mathrm{mn}})}\right|$$

Denote the standard deviations of the respective features over the entire dataset. The SIFT algorithm used in the similarity measure. They are used for feature normalization purposes.

7. Experimental Setup

OVA SVM models using Gaussian kernels as are used both to learn texture signatures and to classify the texture instances in the final feature space obtained after k iterations. OVA SVM models are trained in this final feature space using the training instances. The remaining test instances obtained are used to evaluate the performance. All data processing was performed using MATLAB R2009b

A. Datasets

To evaluate the effectiveness of our approach, we selected thirteen texture images obtained from the standard brodatz dataset. Before being digitized, each of the 512 X 512 texture images were rotated at different degrees. To test the rotation-invariance, and scale-invariance of the method, three different image datasets were generated: non-distorted, rotated, and scaled. The non-distorted image dataset was constructed just from input textures with no rotation and scale changes. Each texture image was partitioned into sixteen 128 X 128 non-overlapping sub images. This dataset comprises 208 different images. The second image dataset is referred to as rotated image dataset, and was generated by selecting the four 128 X 128 innermost sub images from texture images at 0, 30, 60, and 120 degrees. A total number of 208 images were generated the (13 X 4 X 4). Finally in the scaled image dataset,the 512 X 512 non-rotated textures were first partitioned into four 256 X 256 non-overlapping images. Each partitioned sub image was further scaled by using four different ranging from 0.6 to 0.9 with 0.1 intervals. This led to 208 scaled images.

B.Retrieval effectiveness evaluation:

In our project, a simulated query is represented by any of the 208 images in a dataset. The relevant images for each query are defined as the 15 remaining sub images from the same input texture. In this context, a total number of 43056(207 X 208) queries were performed in each dataset. The retrieval effectiveness was measured in terms of relevant retrieval average rate, i.e., the percentage of relevant images among the top N retrieved images.

7. RESULTS

Rotation- invariant representation’s achieved by computing the dominant orientation of the texture images followed by feature alignment. . Initially, take a various train images (data samples) then extract the various features from that rotational texture images and stored in data base. Similarly test the images, then extract the features of text images and compare with data base based similarity features we can extract image (similar) from the data base.sFinally, rotation-invariance is obtained by shifting circularly featured elements within the same scales, so that first elements at each scale correspond to dominant orientations.

8. Conclusion

Three series of experimental were conducted to evaluate the retrieval effectiveness our method. In the first ones, we evaluate the discriminating power of conventional steerable pyramid decomposition in characterizing texture images, and retrieval effectiveness is affected by the presence of scaled and rotated versions of texture patterns. The second and third series of experiments are used to evaluate the rotation, and scale- invariant properties of our approach. Comparisons with the conventional pyramid decomposition and with a recent proposal for rotation, and scale- invariance texture retrieval based on Gabor Wavelets are futher discussed.

REFERENCES

[1] A. Depeursinge, A. Foncubierta-Rodríguez, H. Müller, and D. Van De Ville, “Rotation-covariant visual concept detection using steerable Riesz wavelets and bags of visual words,” Proc. SPIE, vol. 8858, pp. 885816-1–885816-11, Sep. 2013.

[2] A. Depeursinge, A. Foncubierta-Rodríguez, D. Van De Ville, and H. Müller, “Lung texture classification using locally-oriented Riesz components,” in Medical Image Computing and Computer Assisted Intervention (Lecture Notes in Computer Science) vol. 6893. Berlin/Heidelberg, Germany: Springer-Verlag, 2011, pp. 231–238.

[3] A. Depeursinge, A. Foncubierta-Rodríguez, D. Van De Ville, and H. Müller, “Multiscale lung texture signature learning using the Riesz transform,” in Medical Image Computing and Computer Assisted Intervention (Lecture Notes in Computer Science) vol. 7512. Berlin/Heidelberg, Germany: Springer-Verlag, 2012, pp. 517–552.

[4] A. Depeursinge, A. Foncubierta-Rodríguez, A. Vargas, D. Van De Ville, A. Platon, P.-A. Poletti, et al., “Rotation-covariant texture analysis of 4D dual-energy CT as an indicator of local pulmonary perfusion,” in Proc. IEEE 10th ISBI, Apr. 2013, pp. 149–152.

[5] M. N. Do and M. Vetterli, “Contourlets: A directional multiresolution image representation,” in Proc. ICIP, vol. 1. 2002, pp. 357–360.

[6] M. N. Do and M. Vetterli, “Rotation invariant texture characterization and retrieval using steerable wavelet-domain hidden Markov models,” IEEE Trans. Multimedia, vol. 4, no. 4, pp. 517–527, Dec. 2002.

[7] D. L. Donoho, “Ridge functions and orthonormal ridgelets,” J. Approx. Theory, vol. 111, no. 2, pp. 143–179, 2001.

[8] N. Doshi and G. Schaefer, “A comparative analysis of local binary pattern texture classification,” in Proc. IEEE VCIP, Nov. 2012, pp. 1–6.

[9] O. Ghita, D. Ilea, A. Fernandez, and P. Whelan, “Local binary patterns versus signal processing texture analysis: A study from a performance evaluation perspective,” Sensor Rev., vol. 32, no. 2, pp. 149–162, 2012.

[10] G. Gonzalez, F. Fleuret, and P. Fua, “Learning rotational features for filament detection,” in Proc. IEEE Conf. CVPR, Jun. 2009,pp. 1582–1589.

[11] Z. Guo, Q. Li, J. You, D. Zhang, and W. Liu, “Local directional derivative pattern for rotation invariant texture classification,” Neural Comput. Appl., vol. 21, no. 8, pp. 1893–1904, 2012.

[12] Z. Guo, Q. Li, L. Zhang, J. You, D. Zhang, and W. Liu, “Is local dominant orientation necessary for the classification of rotation invariant texture,” Neurocomputing, vol. 116, pp. 182–191, Sep. 2013.

[13] Z. Guo, L. Zhang, and D. Zhang, “A completed modeling of local binary pattern operator for texture classification,” IEEE Trans. Image Process., vol. 19, no. 6, pp. 1657–1663, Jun. 2010.