THE MASK TUNING SCHEME FOR TEXTURE FEATURE EXTRACTION AND ITS PARALLEL IMPLEMENTATION

J. You[1,*] *H.A. Cohen*[2] *E. Pissaloux*[3]

[1]School of Computer & Information Science
University of South Australia, The Levels
South Australia, 5095

[2]Department of Computer Science and Computer Engineering
La Trobe University, Bundoora, Victoria, Australia, 3083

[2]Institut d'Electronique Fondamentale
Université Paris XI
Orsay Cedex 91405, France

ABSTRACT

This paper describes the parallel extension of the mask tuning scheme for texture feature extraction. Unlike other parallel systems in which specific parallel computer architectures are required, our parallel solution to increase processing speed is examined by introducing the concept of remote procedure call(RPC) in distributed systems for local or remote interprocess communication. Based on our previous sequential implementation of the dynamic texture feature extraction associated with a texture 'tuned' mask, the rotational and scale invariant texture classification system is consequently established in parallel by sharing and transferring information on local machines via the network.

Key words and phrases: Feature extraction, mask tuning, texture classification, image segmentation, multiscale, multi-orientation, texture energy, convolution, parallel implementation, remote procedure call (RPC).

1. INTRODUCTION

The analysis of texture is an important consideration in the development and application of computer vision system since visual texture is a fundamental property identifying the surface structure in scenes. However, the characterization of complex texture, necessary for both image segmentation and texture classification, remains a difficult and challenging problem. Traditional approaches to texture analysis involves extracting texture features to constitute a feature space and then performing stochastic search within this feature space to determine a complex classifier[6]. The aim of feature extraction is to represent an image by a set of numerical 'features' so as to remove redundancy from the data and reduce the features dimension. For the classification to be useful, it must be reliable and computationally attractive, which means the choice of the textural features or the models must be as compact as possible, and yet as discriminating as possible.

Structural and statistical approaches can be both applied to extract texture features. The former represents the texture as composite with defined relationships between the primitives, while the latter is represented by a variety of techniques, including pointwise statistical analysis and first and second order statistical properties. Classic methods for texture feature extraction aimed to capture human-like features by some measures such as coarseness and directionality. Most of the new techniques involve statistical applications.

Unlike the usual approaches involving feature extraction and classification rule construction, the special interest of our proposed method is to determine a feature, the texture energy computed using texture 'tuned' masks to directly function as a classifier in a single stage by extending both Laws' texture energy concept[5] and Benke *et al*'s mask tuning scheme[7]. The simplicity and roubustness of computation of the derived features of rotated and/or scaled textures by means of our mask tuning scheme has been reported in our previous papers[9],[15]. In this paper the parallel extension of the mask tuning scheme by using remote procedure call (RPC) in distributed systems is presented. The paper will be organized as following: Section 1 is the introduction which summarizes the existing techniques of texture feature extraction. Section 2 highlights the mask tuning scheme to obtain texture energy via tuned mask as texture feature measurement for classification and segmentation of textured images. Section 3 details the parallel implementation procedures using remote procedure call (RPC) and Section 4 presents the experimental results and the general conclusion is drawn in Section 5.

2. THE MASK TUNING SCHEME

Texture features are a class of image features often used for classification or for segmentation of the image into differently textured regions. The simplicity and robustness of determination of rotation and scale invariant texture features by introducing the texture energy computed using texture 'tuned' masks has been reported in our previous papers[7], [13]. Such a mask tuning scheme is aimed to improve the classification accuracy by adjusting the mask coefficients on the texture training samples for the optimal response to the given figure of merit. The mask is assumed 5*5 size with zero sum in each row. Thus

*This work was partly supported by 1994 French Foreign Ministry International Postdoctoral Fellowship

there are 20 mask coefficients to be determined during the training session. In our approach the local variance after convolution is well-approximated the sum of squared values of convolved image within the test window, which is expressed as below:

$$TE(i,j) = \frac{\sum_{W_x} \sum_{W_y} (I * A)^2{}_{rs}}{P^2 W_x W_y}$$

where the rs sum is over all pixels within a square window W of size $W_x * W_y$ centered on the pixel at i, j, A is a zero sum 'tuned' 5*5 convolution mask and P is the parameter normalizer $P^2 = \sum_{i,j} (A_{i,j})^2$.

The procedure to generate the adaptive mask is shown in Fig. 1.

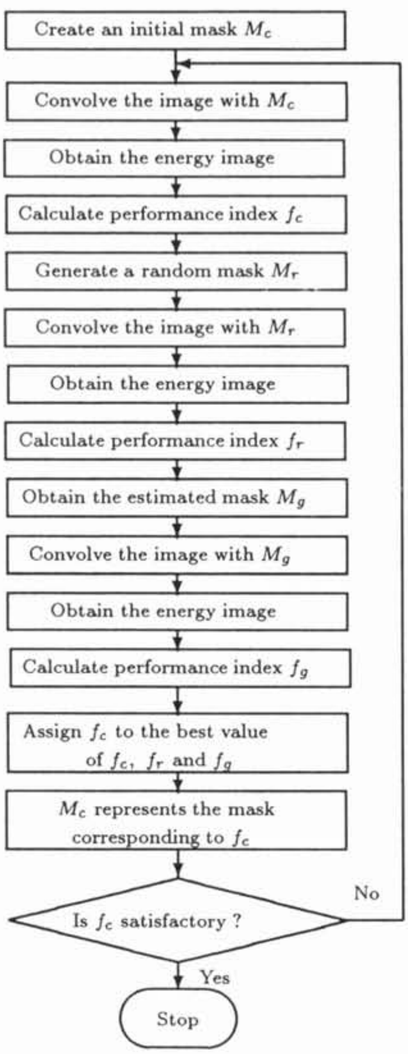

Fig. 1 Main steps in generation the 'tuned' mask

3. THE PARALLEL IMPLEMENTATION

The sequential implementation of the mask tuning scheme reported in our previous publications[7], [13] was an extension of the work of Laws'[3] and Benke *et al*'s[5], which demonstrated satisfactory segmentation of as 15 distinct textures using a single mask. It is noted that both data parallelism and functional parallelism can be applied to achieve speedup in our mask tuning scheme. More specifically, several operators such as different masks can be applied to the image simultaneously for the search procedure to determine the best suitable mask corresponding to the optimal figure of merit; meanwhile the essential operations in the training session, such as the convolution of the image with the given mask and the calculation of the texture energy for the texture image, are implemented by exploiting data parallelism on different regions of the image. Such parallelism can be summarised as below:

- (a) parallel image convolution

  The mask considered is 5*5 size and the operation is performed locally on the image. Therefore, the image can be divided into sub-images with reasonable sizes and the convolution operation can be performed on each sub-images in parallel to speed up the process. In general, the two-dimensional convolution of the image I(i,j) and mask A(i,j) with size 2a+1 by 2a+1 is given by the relation

  $$F(i,j) = A(i,j) * I(i,j)$$
  $$= \sum_{k=-a}^{a} \sum_{l=-a}^{a} A(k,l) I(i+k, j+l)$$

  for i=0,1, ..., M-1 and j=0,1, ..., N-1, where the size of the image is M*N.

- (b) parallel texture energy calculation

  The texture energy is the variance of the convolved image over a local window (9*9 size for training, 15*15 size for final classification and segmentation). Such a local operation can also be realized in parallel to increase the computation speed. The so-called texture energy image can be produced by the following local calculation

  $$E(i,j) = \frac{\sum_{W_x} \sum_{W_y} F^2(i,j)}{P^2 W_x W_y}$$

  where W_x, W_y indicate the window size and P refers to the mask normalizer.

- (c) parallel guided search procedure

  The guided random search algorithm developed for the sequential system is based on the combination of random search and the gradient search, which includes the random guess, current best estimation and the guided estimation. The operation involved in each steps can be divided into frequently executed sub-tasks such as image convolution, texture energy image creation and performance index calculation. These sub-tasks can be implemented in parallel.

In contrast to the conventional parallel implementation in which the mesh or pipeline computer architectures have to be developed to dedicate to the particular task, the concept of remote processing call(RPC) in distributed systems is introduced to simulate and verify the parallel implementation of our mask tuning scheme. In our initial test, there are four servers running by registering with portmapper to look for relevant RPC calls. Each sever then sits and waits until some function-calls are received for related processing. It also sends reply to the client(master) on the completion of each call. The RPC Library enables us to write application programs consisting of a set of procedures that do not all reside on a single computer but on different computers that are interconnected by a communication network. Thus resources available to an application are no longer limited to a single computer and computing power can be added incrementally to the network. The function calls in our test include:

- random mask creation
- image convolution
- texture energy calculation
- performance index calculation

4. EXPERIMENTAL RESULTS

The texture used in this study were selected from Brodatz's well-known compilation. Figure 2 shows 15 unrotated and unscaled histogram equalized texture samples. In order to show the advantages of the mask tuning scheme in capturing the common texture feature invariant of rotation and scale effects on real textures, Figure 3 provides examples of multi-scale and orientation image segmentation by pixel labeling in terms of its texture energy extracted by the tuned mask. Both sequential and parallel implementation of texture classification using the mask tuning scheme are compared. It is noted that the most part of execution time was due to the training session to search for the mask with the optical performance criteria for texture classification. The accuracy and efficiency of the segmentation procedure can be dramatically increased once such a 'tuned' mask is determined via the tuning scheme. As convolution operation plays a key role during mask tuning, the spped-up of the calculation will benefit the overal performance. On the average the sequential execution time for each convolution with 5*5 mask on 256*256 image implemented on Classic SPARC workstation is about 5.8 sec. while the parallel computation with 4 processes increases the processing speed by 15% to 4.96 sec. for each convolution. The higher speed is expected when more processes are invoked for parallel implementation.

5. CONCLUSION

Our conclusion is that a rotational and scale invariant texture classification system can be established by using texture energy via tuned mask as texture feature measurement. The processing speed can be increased by introducing remote procedure call (RPC) for the parallel implementation of the tuning procedure without any special requirement for the computer hardware architecture. The number of processes depends on the throughput of the network (thus on the number of networked computers as well) among which will share and transfer image data. The speed can be further increased by considering the dynamic load balancing requirements of the system for parallel implementation.

6. REFERENCES

[1] G. Fox, M. Johnson, G. Lyzenga, S. Otto, J. Salmon and D. Walker, "Solving problems on concurrent processors", *Englewood Cliffs, Prentice-Hall*, 1987.

[2] P. Lennie, C. Trevarthan, D. Van Essen and H. Wassie, "Parallel processing of visual information" in *Visual Perception: The Neurophysical Foundations* by L. Spillman and J. Werner(eds), Academic Press, New York, 1990.

[3] K.I. Laws, "Textured Image Segmentation", *Ph.D Thesis*, University of Southern California, January 1980.

[4] R.M. Haralick, "Statistical and structural approaches to texture", *Proc. IEEE, 67*, pp. 786-804, 1979.

[5] K.K. Benke, D.R. Skinner and C.J. Woodruff, "Convolution operators as a basis for objective correllates for texture perception", *IEEE Trans. Syst., Man, Cybern.*, Vol. SMC-18, pp. 158-163, 1988.

[6] T.M. Caelli and D. Reye, "On the classification of image regions by colour, texture and shape", *Pattern Recognition*, Vol. 26, No. 4, April, pp. 461-470, 1993.

[7] H.A. Cohen and J. You, "A multi-scale texture classifier based on multi-resolution 'tuned' mask", *Pattern Recognition Letters*, Vol. 13, Number5, 1992, pp. 599-604.

[8] R.L. Kashyap and A. Khotanzad, "A model based method for rotation invariant texture classification", *IEEE Trans. Pattern Anal. Machine Intell.*, Vol. PAMI-8, pp. 472-481, 1986.

[9] A. Khotanzad and J.H. Lu, "Classification of invariant image representations using a neural network", *IEEE Trans. Acoustics, Speech, and Signal Processing*, Vol. ASSP-38, pp. 1028-1038, 1990.

[10] M. Tuceryan and A.K. Jain, "Texture analysis" in *The Handbook of Pattern Recognition and Computer Vision* by C.H. Chen, L.F. Pau and S.P. Wang(eds.), World Scientific Publishing Co., 1992.

[11] S. Peleg, J. Naor, R. Hartley and D. Avnir, "Multiple resolution texture analysis and classification", *IEEE Trans. Pattern Anal. Machine Intell.*, Vol. PAMI-6, pp. 518-527, 1984.

[12] M. Pietikäinen, A. Rosenfeld, and L.S. Davis, "Experiments with texture classification using averages of local pattern matches", *IEEE Trans. Syst., Man, Cybern.*, Vol. SMC-13, pp. 421-425, 1983.

[13] J. You and H.A. Cohen, "Classification and Segmentation of Rotated and Scaled Textured Images Using Texture 'Tuned' Masks", *Pattern Recognition*, Vol. 26, No. 2, Feb., pp. 245 - 258, 1993.

[14] J. You, E. Pissaloux and H.A. Cohen, "A dynamic analysis system for textured images using multiprocessors", to be appeared in the Proc. of *IEEE conference on Massively Parallel Computing Systems (MPCS)*, Ischia, Italy, May 2-6, 1994.

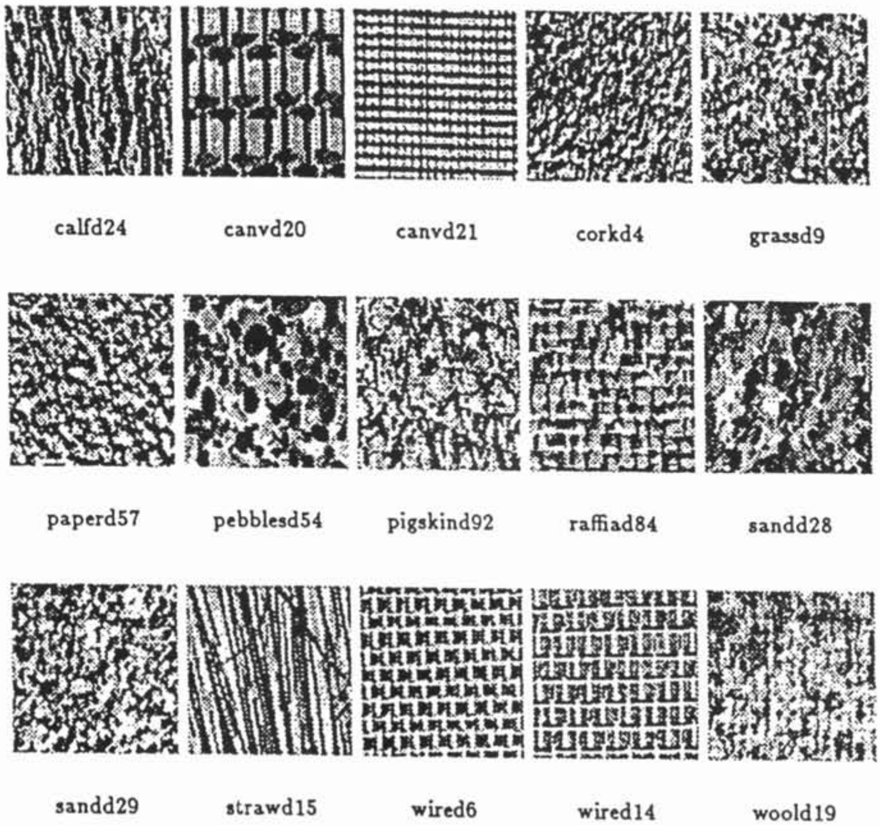

Figure 2. 15 unrotated and unscaled texture samples

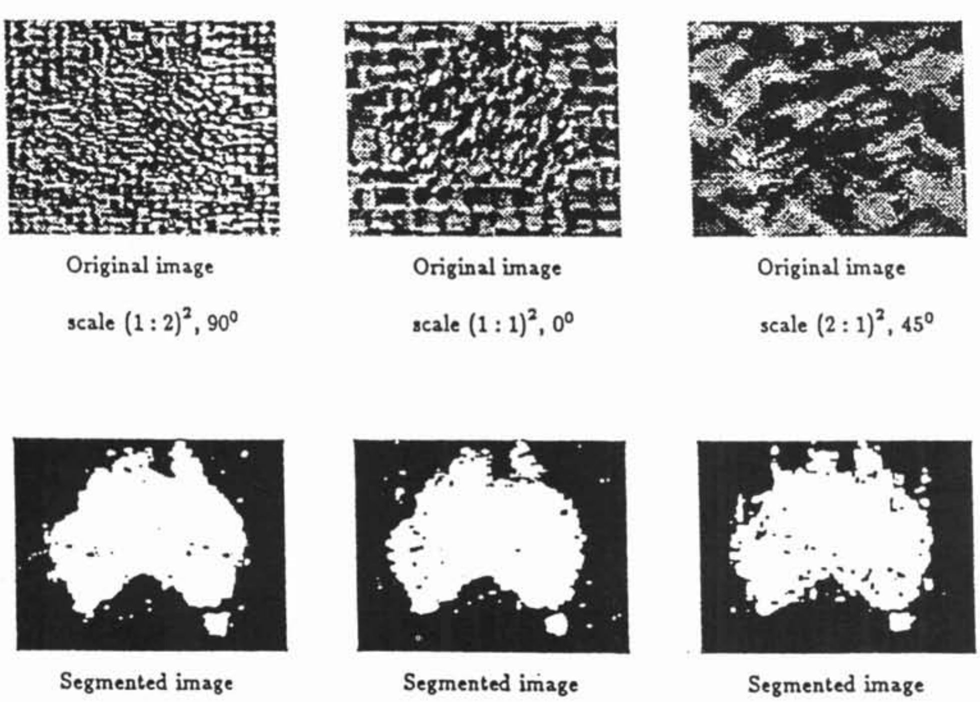

Figure 3. Examples of multi-scale and orientation image segmentation by pixel labelling