*Journal of* ELECTRICAL ENGINEERING, VOL. 56, NO. 1-2, 2005, 36–40

# REGION–BASED TEXTURE CODING AT VERY LOW BITRATES

**Anton Březina** [*] — **Jaroslav Polec** [*] — **Jarmila Pavlovičová** [*]
**Ivana Kolingerová** [**] — **Peter Bandzi** [*]

Our approach to image region approximation offers a complete scheme consisting of few steps. The original image is segmented using two algorithms. An unsupervised colour — texture regions segmentation method (JSEG) and feature distributions based method. Following polygonal approximation of the created regions causes the region boundaries degradation. This approximation has no significant influence upon the image quality and allows increasing the code efficiency. Especially for very low bitrates image coding we achieved better values of objective quality criteria (PSNR). For texture approximation we use the 2D shape independent orthogonal transform (DCT II.). The encoding and decoding of polygons is very efficient because we need to store only their endpoints. The texture is coded with a modified code similar to JPEG arithmetic code.

K e y w o r d s: texture, segmentation, image compression, region-based coding

## 1 INTRODUCTION

Lately, great concern in image processing has been devoted to region-oriented methods. Region oriented image representation offers several advantages over block-oriented approach, *eg* adaptation to the local image characteristics. New algorithms are necessary for image coding if we work with arbitrarily shaped image regions, called segments, instead on rectangular blocks. The original approach for the coding of arbitrarily shaped image segments based on a generalized orthogonal transform was discussed in [1]. An application scheme with cosine transform is proposed in [2]. Two approaches to finding the region boundaries are used in this article, one using the gradient method and the second one based on feature distributions. The boundaries are approximated by polygonal lines, *eg* [4], and the texture inside each region is approximated using basis functions of 2D shape independent orthogonal transform defined on the rectangle circumscribing the given image segment. Each segment is iteratively approximated until the stated quality criterion is met (PSNR, MSE, ... ). This may result in using different numbers of basis functions for each coded segment. The results are linearly quantized and coded with a variable- length code. In our contribution, we developed a new scheme for the approximation and coding of segmented texture images at very low bitrates.

## 2 IMAGE SEGMENTATION

### 2.1 JSEG method

The first approach to image partitioning was based on an unsupervised segmentation method for colour-texture regions [5].

This method does not attempt to estimate the specific model for a texture region. Instead, it tests for the homogeneity of a given colour-texture pattern, which is computationally more feasible than model parameter estimation. In order to identify its homogeneity, the following assumptions about the image are made:

- Each image contains a set of approximately homogeneous colour-texture regions. This assumption conforms to our segmentation objective.
- The colour information in each image region can be represented by a set of few quantized colours.
- The colours between two neighbouring regions are distinguishable — it is a basic assumption in any colour image segmentation algorithm.

The segments are found in a few steps. First, colours in the image are quantized to several representative classes and then image pixels are replaced by their corresponding colour class labels. This way we obtain a class-map. The class-map can be viewed as a special kind of texture composition. In the class-map the so-called $J$-values are solved from the local neighbourhood of a pixel. These $J$-values correspond to the minimum variation of texture in image regions. The measure $J$ is defined as

$$J = SB/SW = (ST - SW)/SW$$

where $SW$ is the sum of all so-called within-class variances of points belonging to distinct colour classes in the image and $ST$ is the total variance of image points in the class map.

The criterion measures the distances between different classes $SB$ over the distances between the members within each class $SW$. A higher value of $J$ indicates that the classes are more separated from each other and the

[*] Department of Telecommunications, Faculty of Electrical Engineering and Information Technology, Slovak University of Technology, Ilkovičova 3, 812 19 Bratislava, Slovakia, E-mails: {brezina}{polec}{pavlovic} bandzi@ktl.elf.stuba.sk

[**] Department of Computer Science and Engineering, Faculty of Applied Sciences, University of West Bohemia in Pilsen, Univerzitní 8, Pilsen, Czech Republic, E-mail: kolinger@kiv.zcu.cz

ISSN 1335-3632 © 2005 FEI STU

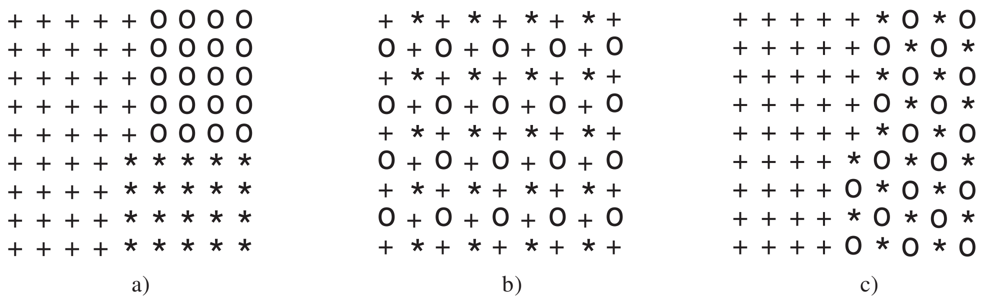


**Fig. 1.** An example of $J$-values for different types of texture maps: a) $J = 1.72$, b) $J = 0$, c) $J = 0.855$.

members within each class are closer to each other, and vice versa.

For the case when an image consists of several homogeneous colour regions, the colour classes are more separated from each other and the value of $J$ is large. On the other hand, if all colour classes are uniformly distributed over the entire image, the value of $J$ tends to be small. The larger the $J$-value, the more likely the corresponding pixel lies near a region boundary.

Examples of a class-map are shown in Fig. 1, where label values are represented by three symbols, '$*$', '$+$', and '0'. Usually, each image region contains pixels from a small subset of the colour classes and each class is distributed in a few image regions.

Finally, a region growing and merging method is applied to the image of $J$-values, the so-called $J$-image, to obtain the final segmentation. $J$-images correspond to measurements of local homogeneities at different scales, which can indicate potential boundary locations. A spatial segmentation algorithm grows regions from seed areas of the $J$-images to achieve the final segmentation.

The $J$-image allows us to use a multiscale region-growing method. Consider the original image as one initial region. The algorithm starts to segment all the regions in the image at an initial large scale. It then repeats the same process on the newly segmented regions at the next smaller scale until the minimum specified scale is reached.

The over-segmented regions after region growing are merged based on their colour similarity. The pair of regions with the minimum distance is merged together. The process continues until a maximum threshold for the distance is reached.

### 2.2 Feature distributions based method

The next approach uses distributions of local binary patterns and pattern contrast for measuring the similarity of adjacent image regions [6] in grey level images. Algorithm consists of three phases: Hierarchical splitting divides the image into regions of roughly uniform texture. Then, agglomerative merging procedure merges similar adjacent regions. Finally, pixel-wise classification is used to improve the boundary localization.

In the first step of the segmentation process the image is divided into square-shaped blocks such that the texture inside each of them meets the stated homogeneity criterion. The joint LBP (Local Binary Pattern) and Contrast distribution are used for measurement of the adjacent textures similarity. By definition, LBP is invariant to any monotonic grey-scale transformation. The LBP/C distribution is approximated by a discrete two-dimensional histogram of size $256 \times b$ (256 — number of grey scale levels, $b$ — number of contrast intervals). Choosing a b value is a trade-off between the discriminative power and the stability of the texture transform. As a homogeneity criterion the statistical test MSE is used. It is a pseudo-metric for feature distribution comparison.

In the second step the algorithm merges that pairs of adjacent segments that have the smallest merger importance. In other words, at each step a certain area is with merger importance criterion (product of MSE and pixel count of smallest area) compared to all adjacent areas. If the smallest difference is bigger than a threshold, merging stops and new merging continues with non- merged areas until all areas have been proceeded. Otherwise the area with the smallest difference is joined to the certain area and new merging continues with the newly created area.

To improve the localization of the boundaries a simple pixel-wise classification algorithm is used as a third step. If the hierarchical splitting and agglomerative merging phases have succeeded, this method could be considered as quite reliable estimates of the different textured regions present in the image. If an image pixel is on the boundary of at least two distinct textures (*ie* the pixel is 4- connected to at least one pixel with a different label), we place a discrete disc with radius $r$ on the pixel and compute the LBP/C histogram over the disc. We compute the MSE distances between the histogram of the disc and the models of those regions, which are 4-connected to the pixel in question. We relabel the pixel, if the label of the nearest model is different from the current label. If the pixel is relabelled, we update the corresponding texture model accordingly, hence the texture models become more accurate during the process. In the next scan over the image only the neighbourhoods of those pixels are checked which were relabelled in the previous sweep. The process of pixel-wise classification continues until no pixels are relabelled or maximum number of sweeps is reached.

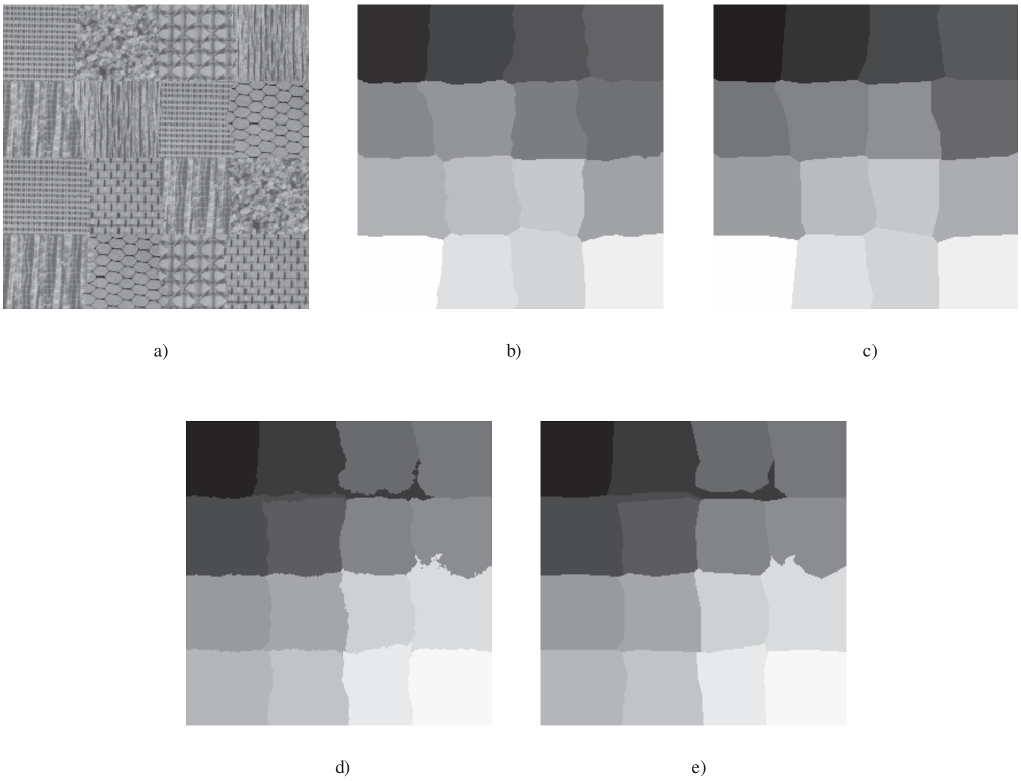


**Fig. 2.** Mixed texture pattern image, 256 × 256 pixels, 256 grey levels: a) original, b) segmented image **a** with JSEG, c) polygonal approximation of **b**, d) segmented image **a**with feature distributions based method, e) polygonal approximation of **d**.

The result of segmentation is an image consisting of regions defined by their unique grey value.

## 3 POLYGONAL APPROXIMATIONS OF THE REGION BOUNDARIES

First, the boundaries of all input regions are to be found. The boundary point is each point, its grey value is equal to the region grey value, but the grey value of at least one of neighbouring pixels differs from the region grey value. To find the boundaries, 8-directional algorithm based on LML (left-most-looking) rule was used [7]. Now each segment is described by its own boundary so that between neighbouring segments there are in all cases two parallel boundaries, one for each segment.

These doubled boundaries between neighbouring segments are reduced. It is pointless to encode both of them. Also very small segments are being attached to their larger neighbours in this step. This has also a significant impact on the resulting code efficiency.

The boundaries of the segments are simply approximated with polygons.

It has been shown [8] that the boundary degradation caused by this step has only a small influence on the quality of the resulting image, but is very important for increasing the code efficiency.

## 4 TEXTURE APPROXIMATION

The texture inside each region is again approximated using basis functions of 2D shape independent orthogonal discrete cosine transform II [2]. The use of wavelet transforms for segmented image compression has been discussed in [3]. Unlike other methods, in our approach the number of basis functions is derived from the quality of approximation requirements. The approximation quality comes up to PSNR 24−25 dB for each segment with respect to the resulting bit rate of the encoded image as we concentrate on the behaviour of the proposed scheme with low bitrates. However, it is possible to interactively approve the quality of approximation or to use a further, more exact approximation. Such a process takes advantage from coding each segment separately, as not all of the textures inside the region boundaries need to be coded with the same number of coefficients. The results are normalized for the human visual system with respect to spectral elements that are most important for human vision. Then the coefficients are linearly quantized. It results in an optimal number of basis functions utilized individually for each region. The number is optimized with respect to desired PSNR.

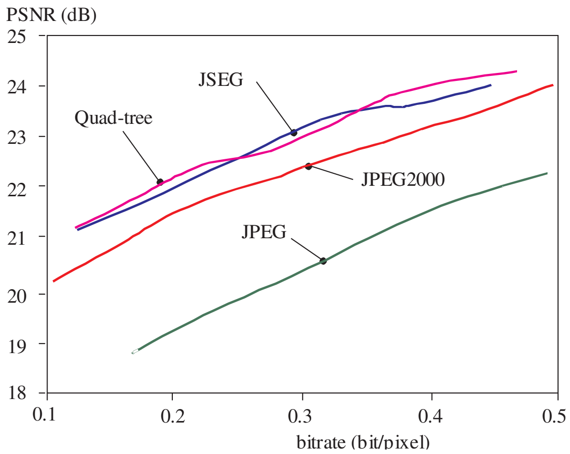



**Fig. 3.** Diagrams of PSNR performance (DCT II texture approximation).

## 5 CODING

### 5.1 Boundary coding

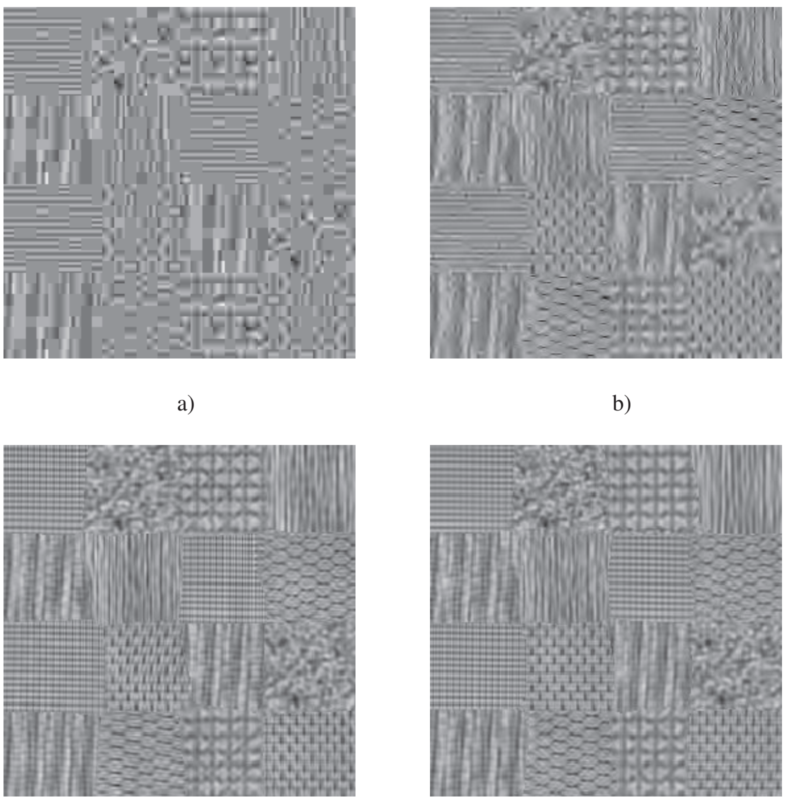
The polygonal approximation supports additional bit rate reduction, while there is no need to encode all directions between the endpoints of a boundary segment. Storing only several selected points between two endpoints is sufficient for further image decoding. Creating the line between two points of a boundary segment in the decoder is based on Bresenham algorithm [9]. Then the boundary is defined without further coding. Finally, the whole data stream consisting of all encoded boundaries is encoded once more, using the modified Huffman code with DPCM.

### 5.2 Texture coding

The final step of the scheme is a binary arithmetic code of the quantized basis functions used for texture approximation. The code used in this step is very similar to JPEG arithmetic code.

## 6 CONCLUSION

The method is very sensitive to the quality of segmentation, number of regions, their sizes and degree (depth)



a) b)

c) d)

**Fig. 4.** Reconstructed images: a) resulting image of JPEG approach (0.2 bit/pixel, PSNR 19.7 dB), b) resulting image of JPEG2000 approach (0.2 bit/pixel, PSNR 21.6 dB), c) reconstructed image processed by proposed method using segmentation map shown on Fig. 1c (0.19 bit/pixel, PSNR 22.01 dB), d) reconstructed image processed by proposed method using segmentation map shown on Fig. 1e (0.22 bit/pixel, PSNR 22.53 dB)

of approximation. Especially, a big size of regions increases the coding time rapidly. Also, a greater number of image regions lowers the efficiency of the proposed method with respect to the bit rate. The comparison with block based approach (classical JPEG) and JPEG 2000 is illustrated in figures. Compressed images via both JPEG and JPEG 2000 were obtained using the VcDemo program [10].

## References


[1] GILGE, M.—ENGELHART, T.—MEHLAN, R.: Coding of Arbitrary Shaped Image Segments Based on a Generalized Orthogonal Transform, Signal Processing: Image Communication **1** (1989), 153180.

[2] KAUP, A.—AACH, T.: Coding of Segmented Images Using Shape-Independent Basis Functions, IEEE Transactions on Image Processing **7** No. 7 (1998), 937–947.

[3] VARGIC, R.—PROCHÁSKA, J.: Wavelet Based Compression of Segmented Images Using Baseline Non-Segmented Approach, Proceedings of IEEE ICIT 2003, Maribor, Slovenia, December 10–12, 2003, pp. 955-958.

[4] EDEN, M.—KOCHER, M.: On the Performance of a Contour Coding Algorithm in the Context of Image Coding. Part I: Contour Segment Coding, Signal Processing **8** (1995), 381–386.

[5] DENG, Y.—MANJUNATH, B. S.: Unsupervised Segmentation Method for ColourTexture Regions in Images and Video, IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI '01), August 2001, http://vision.ece.ucsb.edu/publications/01PAMIJseg.pdf.

[6] OJALA, T.—PIETIKÄINEN, M.: Unsupervised Texture Segmentation Using Feature Distribution, Pattern Recognition **32** (1999), 477–486, http://www.ee.oulu.fi/mvg/publications/show_pdf.php?ID=16.

[7] GONZALEZ, R. C.—WINTZ, P.: Digital Image Processing. Second ed., Reading. Addison-Wesley Publishing Company, Tokyo, 1987, p. 503.

[8] POLEC, J. et al.: New Scheme for Region Approximation and Coding with Shape Independent Transform, PHOTOGRAMMETRIC COMPUTER VISION 2002, Graz, 2002, 19 Posters of WG 8 "Reliability and Performance of Algorithms".

[9] FERKO, A.—RUŽICKÝ, E.: Computer Graphic, Sapienta, Bratislava, 1995, pp. 58–60. (in Slovak)

[10] Information and Communication Theory Group of TU-Delft: VcDemo (August 2002, ver. 4) — Image and Video Compression Learning Tool, http://www-it.et.tudelft.nl/ inald/vcdemo/(2004-05-20).


Received 26 May 2004


**Anton Březina** is with the Slovak University of Technology, Faculty of Electrical Engineering and Information Technology.

**Jaroslav Polec** is associate professor at the Slovak University of Technology, Faculty of Electrical Engineering and Information Technology.

**Jarmila Pavlovičová** is with the Slovak University of Technology, Faculty of Electrical Engineering and Information Technology.

**Ivana Kolingerová** is associate professor at the Department of Computer Science and Engineering, Faculty of Applied Sciences, University of West Bohemia in Pilsen.

**Peter Bandzi** is with the Slovak University of Technology, Faculty of Electrical Engineering and Information Technology.