A Novel Technique of Error Concealment Method Selection in Texture Images Using ALBP Classifier

Želmíra TÓTHOVÁ, Jaroslav POLEC

Dept. of Telecommunications, Slovak University of Technology, Ilkovičova 3, 812 19 Bratislava, Slovakia

polec@ktl.elf.stuba.sk, florekova.zelmira@gmail.com

Abstract. *There are many error concealment techniques for image processing. In the paper, the focus is on restoration of image with missing blocks or macroblocks. Different methods can be optimal for different kinds of images. In recent years, great attention was dedicated to textures, and specific methods were developed for their processing. Many of them use classification of textures as an integral part. It is also of an advantage to know the texture classification to select the best restoration technique. In the paper, selection based on texture classification with advanced local binary patterns and spatial distribution of dominant patterns is proposed. It is shown, that for classified textures, optimal error concealment method can be selected from predefined ones, resulting then in better restoration. For testing, three methods of extrapolation and texture synthesis were used.*

Keywords

Error concealment, re-synthesis, inpainting, texture, extrapolation, classification.

1. Introduction

The transmission of images coded by block or segment based techniques via a noisy channel [5] may lead to block or segment loss. Therefore error detection and concealment at the decoder side has to be applied. Commonly, spatial error concealment is used. It utilizes the surrounding correctly received image information to restore the damaged or missing pixels. A standard approach [4] assumes that the image content is changing smoothly. Hence the algorithm tries to restore the transition across the block boundary as smooth as possible. The extrapolation-based method of [25] tries to reconstruct the missing pixels as a weighted linear combination of correctly received pixels. Hence, the method is computationally very complex. The transmission of block-coded image data via wireless channel is described in [6]. Very interesting and novel method for spatial error concealment based on successive extrapolation of missing blocks is described in [7]. Method for error concealment using neural networks for block-based image coding is described in [26].

The texture of an image might be defined broadly as the interrelationships between pixels in that image. The ability to analyze and manipulate image texture has a number of interesting applications. The simplest application is to create a new image with the same texture but of different size and shape to a sample image. Seamless editing of images is also a possibility. For example an object could be removed from an image by synthesizing a new section of the background texture over the top of it. These applications all rely on the ability to re-synthesize a sample texture to fit a variety of constraints.

In automated systems, it is difficult to decide which of the methods offered would be the most successful one. As selection method, one of the texture classification methods is proposed here.

The Brodatz texture database [10], VisTex [11] and Simoncelli personal database [12] were used as a source.

2. Synthesis

2.1 Non-hierarchical Procedure for Re-synthesis of Complex Textures

Procedure for synthesis of an image with the same or similar texture as a given input image was proposed by Harrison in [3]. To achieve this, the output image is built up by successively adding pixels selected from the input image. Pixels are chosen by searching the input image for patches that closely match pixels already present in the output image. It is shown that the accurate reproduction of features in the input texture depends on the order in which pixels are added to the output image. A procedure for selecting an ordering which transfers large complex features of the input to the output image is described. This procedure is capable of reproducing large features even if only the interactions of nearby pixels are considered.

2.2 Discrete Orthogonal Transforms for Gappy Image Extrapolation

Papers [1], [2], and [8] present different methods for approximation of non-square areas. The texture of an area is successively approximated and then cut to the shape of the segment. The same principle is used in this paper for error concealment by estimating the missing image content using the surrounding area belonging to the same object, thus (supposedly) having the same texture pattern. The image content of the known segments is successively approximated using DCT II [9] and the missing segment is obtained by extrapolation [25] (also [7]).

2.3 Object Removal by Exemplar-Based Inpainting

Algorithm is proposed for removing large objects from digital images. The challenge is to fill in the hole that is left behind in a visually plausible way. In the past, this problem has been addressed by two classes of algorithms: 1. “texture synthesis” algorithms for generating large image regions from sample textures, and 2. “inpainting” techniques for filling in small image gaps. The former work well for “textures” – repeating two dimensional patterns with some stochastic; the latter focus on linear “structures” which can be thought of as one-dimensional patterns, such as lines and object contours.

[6] presents algorithm that combines the advantages of these two approaches. Exemplar-based texture synthesis contains the essential process required to replicate both texture and structure; the success of structure propagation, however, is highly dependent on the order in which the filling proceeds. Authors propose a best-first algorithm in which the confidence in the synthesized pixel values is propagated in a manner similar to the propagation of information in inpainting. The actual color values are computed using exemplar-based synthesis. Computational efficiency is achieved by a block based sampling process.

3. Classification

In this part, the requirements for classifier selection will be defined and the classifier will be selected.

Invariance towards external effects as rotation, scaling, illumination, etc. are usually not required for similar applications. Naturally, a model with some of these characteristics will be preferred, if it does not result in substantial increase of time or memory demands of the algorithm. Computational complexity will be one of the important factors. The most important attribute for decision will be the success rate of classification.

Since for the classification algorithm here, usually no requirements are strictly defined, we will select the model that solves the problem as good as possible, which is the representation that is computationally effective as well as it guarantees high success rate of the classification. Invariance features are naturally an advantage.

If we consider conclusions from [13, 14, 15, 16, 17, 20, 21, 19], the best results from the success rate point of view were acquired using Markov random fields (in particular, VZ-classifier), local binary patterns and advanced local binary patterns with spatial distribution of dominant patterns. Success rate of the classification approached 100%. From other methods, model for DCT coded images could be mentioned. This representation should be used in applications, where DCT coded images should be classified (as JPEG, for example). In this case it is a great advantage compared to other methods, that it does not require conversion to bitmap format, which saves time for processed image. For photorealistic images, PCA and neural networks [18] are successfully used for classification.

Local binary patterns (LBP) have one great advantage compared to VZ-classifier. It is their invariance feature regarding rotation and histogram equalization. Rotation invariance results from the used operator, which, in turn, generates computational requirements increase, because for each computed local binary pattern, the one with the lowest value must be found, which means to do bit shifting. Invariance towards histogram equalization comes directly from the definition of LBP operator. If for two neighboring pixels i and j holds $x_i \neq x_j$, where x_i is luminance for pixel i, this inequality holds also after histogram equalization. And, as LBP operator is based on these inequalities, histogram equalization has no influence. On the contrary, it is mentioned in [24], that for using VZ-classifier in preprocessing phase, it is necessary to do image luminance normalization to get invariance towards linear variations of the luminance within all images.

Therefore, invariance towards rotation or histogram equalization for LBP would not bring increase of algorithm time demand. Another advantage of LBP compared to VZ-classifier is, that components of feature vector for LBP are known before, on the other hand, for VZ-classifier are acquired using k-means clusterisation in training phase, which slows the process down.

If we decide to use local binary patterns for representation, either local binary patterns as used by Ojala [20] (LBP), or advanced local binary patterns with spatial distribution of dominant patterns from Liao and Chung [14] (ALBP) might be employed. The difference between them is using the spatial distribution of dominant patterns. To create representing grey level aura matrix is time and memory consuming operation, but it can bring significant increase of probability of correct classification. The decision is, in this case, very subjective. For this research, ALBP classifier was selected. Computational complexity can be decreased using only one neighborhood. In [20], LBP was tested with more types of LBP operator. In some cases, combination of more neighborhood types was used, leading to success rate increase. In [14], more than 99% of textures were correctly classified. Unfortunately, the paper does not specify what type of the neighborhood was used. LBP model using 3 x 3 neighborhood reached success rate slightly over 80%. It should be mentioned, that certain improvement can be brought by feature of advanced local binary patterns, that is, to create a feature vector, where not only uniform patterns are used, but also 20 of the most frequent (dominant) from all patterns.

Feature vector plays a central role in the algorithm. It is a representation of the texture describing the texture characteristics as accurately as possible. Required property of the feature vector is, that it should be specific for each texture class. It means that, in Euclidean space, vectors of textures from one class should be close to each other and vectors from textures from different classes should be further from each other. Our feature vector will be ALBP representation. To acquire a model for the image, the procedure is: For each pixel, its value is computed. It is saved into the pattern matrix with dimensions same as input image (without margins for which LBP is not computed). At the same time, occurrence frequencies for the patterns are saved. After going through all pixels we define 20 most frequently used patterns in the texture, other counts are reset to zero. Number of occurrences of the most frequent patterns is used as a factor to divide the number of occurrences of each pattern, which gives us the probability of this pattern occurrence in the image (more precisely – probability of occurrence with regard to dominant patterns).

Pattern matrix shall be modified as well. If the matrix includes the pattern belonging to 20 of the most frequent patterns in the image, value there is 1, otherwise it is 0. From the modified matrix, luminance matrix of aura is created, having 4 components that are normalized afterwards.

Feature vector will be created using probability values pertaining to local binary patterns. They shall be ordered by LBP values. Values from normalized aura matrix will be added to the vector. Overall, the feature vector will have a dimension 40 (36 unique local binary patterns + 4 components of aura matrix).

Before starting the classification, database of predefined classes is created. In machine learning, this is called supervised learning. K textures are selected from each class. Number of textures must remain the same for each class. For each of the textures, a model (feature vector) is created, ALBP in our case. This model, together with class label that it belongs to, is saved into the database. In this way we get the database for m classes, that contains $m \times k$ unique vectors. The system is then ready for classification.

Classification procedure: Input image shall be categorized into one of the classes defined in training phase. As first, ALBP model is created for the image. To find the correct class, "*k nearest neighbors*" classifier is used. It is suitable for linearly inseparable classes. Two classes are linearly inseparable in n-dimensional space, if there exists (n-1)-dimensional super plane dividing these classes. It is evident that the class pairs from training phase may not have this feature, and they actually often do not have it.

Algorithm "k nearest neighbors" is: Input image feature vector distance from database vectors is found, where Euclidean metrics is used. From trained database, k vectors nearest to input image are selected. The image is assigned to class most often present.

4. Solution

To select a method for error masking in textured images, we have two possibilities.

1. Classify a damaged texture correctly – define a class, for which one of the masking methods is substantially more effective compared to others, and proceed as [23].
2. Use all available error masking methods and use corrected images as classifier training. Corrected image with one classification method is a class. For testing, undamaged part of the image is used. Corrected image is classified based on the classification of undamaged part.

Training on Brodatz texture database [10], procedure (1) results were very unconvincing, therefore we decided to use procedure (2) (Fig. 1). Software classifier [22] was used, that is implementation of [14].

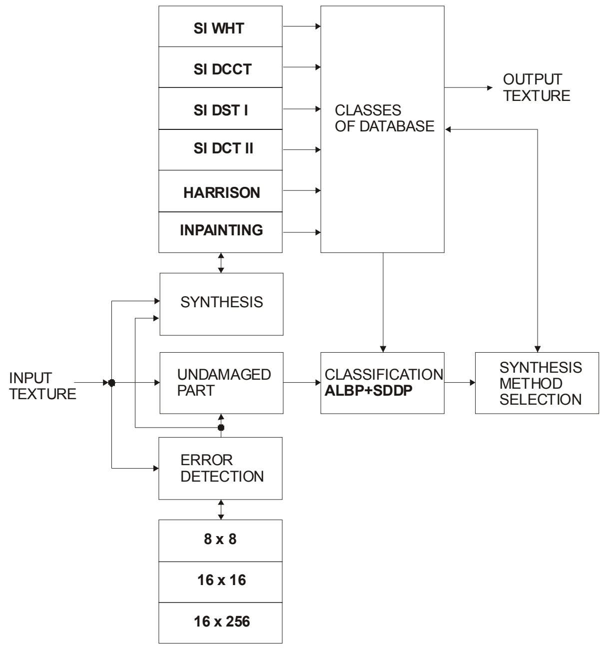


Fig. 1. Masking method selection procedure. Input texture is classified by its undamaged part (ALBP-advanced local binary patterns, SDDP-spatial distribution of dominant patterns), suitable synthesis method is selected accordingly (SI -shape independent transform) and applied to recover the erroneous or missing part.

5. Results

Depending on the way how the texture images were acquired, they can be divided into natural (photographic) and artificial images. Photographic texture images are mostly natural textures (grass, clouds, water, etc.), pictures of fabric textures, patterns, materials, etc. One of the most common divisions of texture images is based on geometric adjustment of inside objects. By periodicity, textures are divided to [12]:

- periodic - mostly artificial texture images generated exactly according to the specified rules (Fig. 2 a);
- pseudo-periodic - artificial textures generated with no specific rules (e.g. by adding boundary tolerance or probability distribution of pattern's appearance) or natural texture images (Fig. 2 c);
- non-periodic - artificial textures with random added patterns (Fig. 2 b, f) or natural patterns with random (Fig. 2 e) or structured (Fig. 2 d) placed patterns.

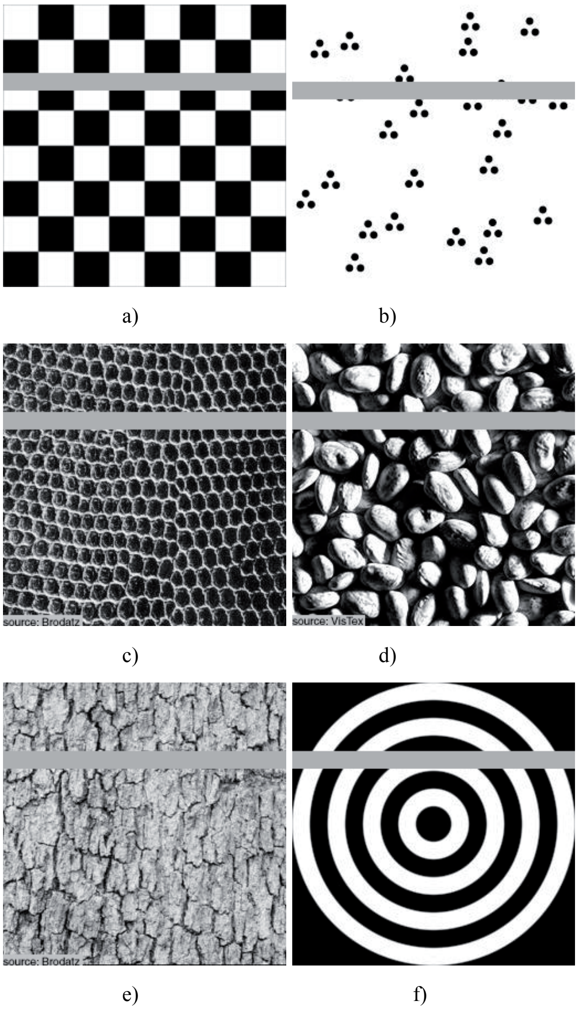


Fig. 2. Damaged images: (a) checkerboard, (b) jdice, (c) D3, (d) grains, (e) bark, (f) bullseye.

When the images are sent through the channel using block-oriented compression techniques, they are error prone. Images received can contain errors. These errors occur as missing individual blocks or a group of blocks. In the experiment we examined both types of errors – missing isolated blocks and group of missing successive blocks. For the block loss, reconstruction techniques for missing 8x8 blocks (as in JPEG coding) or 16x16 macroblock (as in video coding) were examined. For group of successive blocks, we can consider loss of blocks across the image, therefore dropped image part of size 16x256 pixels.

Tab.1 shows the results on selected textures from each of the mentioned classes when using three listed masking approaches and three error types. ALBP was used for classification. The images are shown in Fig.2 - Fig.8.

missing 8x8 block		DCT	Harrison	Inpainting
checkerboard	PSNR	42.21	infinity	infinity
	ALBP		X	
jdice	PSNR	13.65	24.24	29.95
	ALBP			X
D3	PSNR	17.00	19.16	13.57
	ALBP	X		
grains	PSNR	15.36	13.79	13.47
	ALBP	X		
bark	PSNR	13.90	13.06	13.63
	ALBP	X		
bullseye	PSNR	36.33	38.94	36.02
	ALBP	X		

missing 16x16 block		DCT	Harrison	Inpainting
checkerboard	PSNR	47.63	38.581	infinity
	ALBP	X		
jdice	PSNR	6.20	6.78	28.13
	ALBP			X
D3	PSNR	13.78	15.49	15.55
	ALBP			X
grains	PSNR	10.10	13.10	12.36
	ALBP		X	
bark	PSNR	11.19	13.05	11.26
	ALBP		X	
bullseye	PSNR	38.44	13.56	30.14
	ALBP	X		

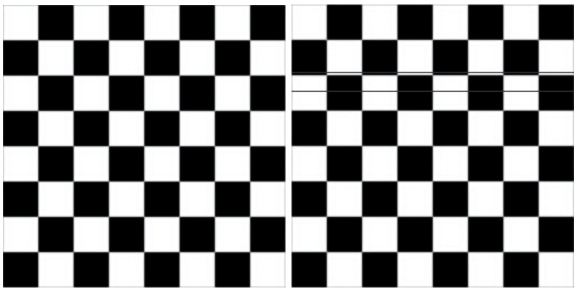
missing 16x256 slice		DCT	Harrison	Inpainting
checkerboard	PSNR	48.46	9.39	34.29
	ALBP	X		
jdice	PSNR	12.78	14.74	10.70
	ALBP			X
D3	PSNR	11.37	11.26	11.20
	ALBP	X		
grains	PSNR	7.71	8.56	8.85
	ALBP			X
bark	PSNR	12.13	12.77	12.27
	ALBP	X		
bullseye	PSNR	36.01	11.19	12.93
	ALBP	X		

Tab. 1. Results on selected textures (X denotes classification result from the 3 reconstructed images, shading means the best result).

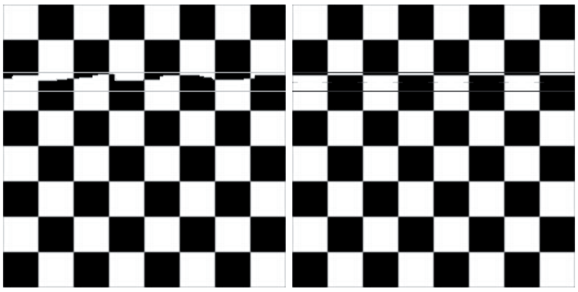
6. Conclusion

Tab. 1 shows, that the results of ALBP classifier are not equivalent to results produced by PSNR in five out of eighteen cases, however, for textures “checkerboard” and „bark“, results produced by PSNR from three masking types are very similar, therefore it can be stated that it does not correspond in three out of fifteen cases.

Paper [27] shows content-based image error detection and error concealment algorithm for improving the image quality degraded during its transmission over wireless channel The damaged image blocks are detected by exploring the contextual information in images, such as their consistency and edge continuity. The statistical characteristics of missing blocks are then estimated based on the types of their surrounding blocks (e.g., smoothness, texture and edge). Finally, different error concealment strategies are applied to different types of blocks in order to achieve better visual quality.

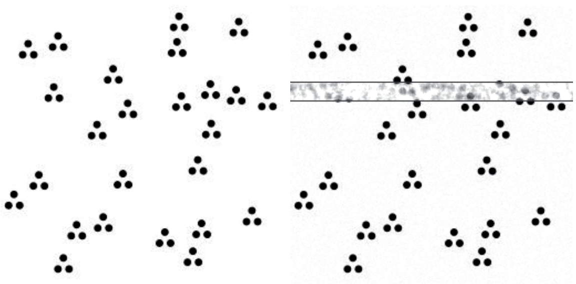


a) b)

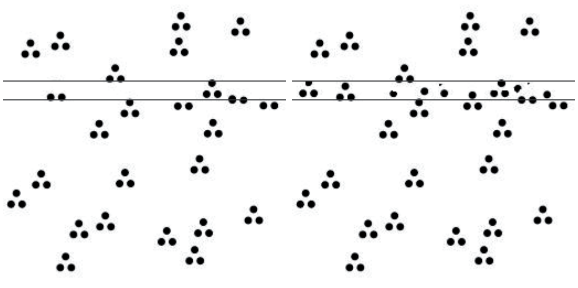


c) d)

Fig. 3. Missing strip of blocks reconstruction: (a) original picture checkerboard and results: (b) transformational extrapolation, (c) re-synthesis and (d) inpainting.

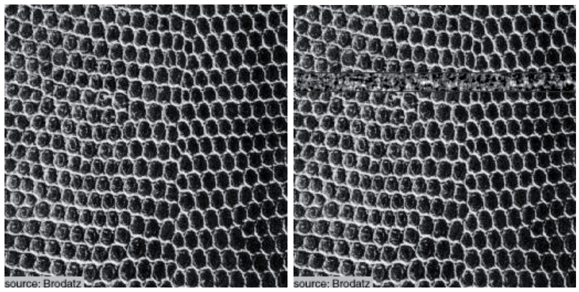


a) b)

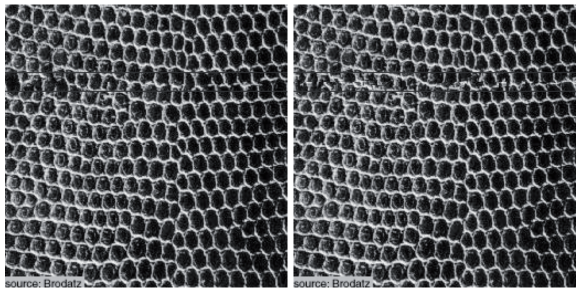


c) d)

Fig. 4. Missing strip of blocks reconstruction: (a) original picture jdice and results: (b) transformational extrapolation, (c) re-synthesis and (d) inpainting.



a) b)



c) d)

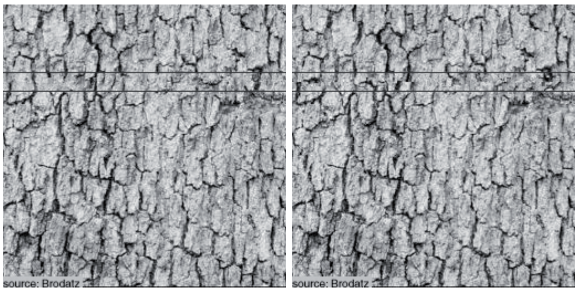
Fig. 5. Missing strip of blocks reconstruction: (a) original picture D3 and results: (b) transformational extrapolation, (c) re-synthesis and (d) inpainting.

a) b)

c) d)

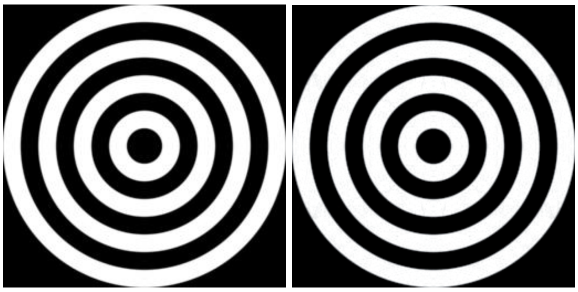
Fig. 6. Missing strip of blocks reconstruction: (a) original picture grains and results: (b) transformational extrapolation, (c) re-synthesis and (d) inpainting.

a) b)

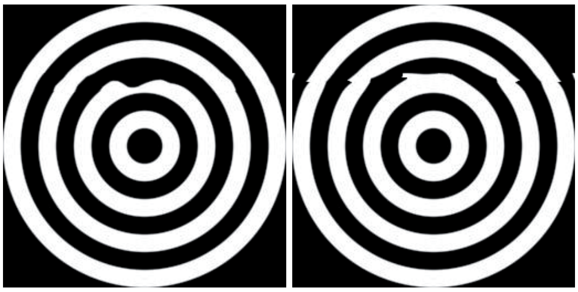


c) d)

Fig. 7. Missing strip of blocks reconstruction: (a) original picture bark and results: (b) transformational extrapolation, (c) re-synthesis and (d) inpainting.



a) b)



c) d)

Fig. 8. Missing strip of blocks reconstruction: (a) original picture bullseye and results: (b) transformational extrapolation, (c) re-synthesis and (d) inpainting.

Our experiment proves that there is no error concealment method that brings generally speaking satisfactory result for all types of textures. The experiment shows, that reverted procedure using the classifier for masking purposes is a suitable method for evaluation of results from different kinds of texture synthesis algorithms. Therefore, the distance of feature vectors can be considered as a suitable objective criterion to substitute MSE or PSNR especially in cases, where these criteria can not be used because the parts of image are missing. As a result we propose this method for selection of masking procedure as suitable one.

Acknowledgements

The research described in the paper was financially supported by the Slovak Research Grant Agency (VEGA) under grant No. 1/0883/08.

References

[1] GILGE, M., ENGELHART, T., MEHLAN, R. Coding of arbitrary shaped image segments based on a generalized orthogonal transform. *Signal Processing: Image Communication*, 1989, no. 1, p. 153-180.

[2] KAUP, A., AACH,T. Coding of segmented images using shape-independent basis functions. *IEEE Trans. on Image Processing*, 1998, vol.7, no. 7, p. 937-947.

[3] HARRISON, P. A non-hierarchical procedure for re-synthesis of complex textures. In *WSCG'2001*. Plzeň (Czech Republic), 2001, p. 190-197.

[4] WANG, Y., ZHU, Q. F., SHAW, L. Maximally smooth image recovery in transform coding. *IEEE Transactions on Communications*, 1993, vol. 41, no. 10, p. 1544–1551.

[5] KOTULIAKOVÁ, K. *Hybrid ARQ Methods in Wireless Communication Channels*. PhD. Thesis, SUT, Bratislava, 2005.

[6] CRIMINISI, A., PÉREZ, P., TOYAMA, K. Object removal by exemplar-based inpainting. In *Proceedings of 2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2003, p. 721-728.

[7] MEISINGER, K., KAUP, A. Spatial error concealment of corrupted image data using frequency selective extrapolation. In Conf. Rec. *IEEE Int. Conf. on Acoustics, Speech, and Signal Processing (ICASSP)*. Montreal (Canada), May 2004, p. III-209-III-212.

[8] POLEC, J. et al. New scheme for region approximation and coding with shape independent transform. In *Proceedings IAPRS*, vol. XXXIV, part 3A/B “Photogrammetric Computer Vision”, Graz, 2002, p. B-214-217.

[9] BESSLICH, PH.W., LU, T., *Diskrete Orthogonal-transformationen*, Berlin: Springer-Verlag, 1990, 312 pages.

[10] BRODATZ, P. *Textures, a Photographic Album for Artists and Designers*. Dover Publications Inc., New York, 1966, http://www.ux.his.no/~tranden/brodatz.html

[11] Vision Texture, http://vismod.media.mit.edu/vismod/imagery/VisionTexture/ vistex.html

[12] PORTILLA, J., SIMONCELLI, E. *Representation and Synthesis of Visual Texture*. http://www.cns.nyu.edu/~eero/texture/

[13] VARMA, M., ZISSERMAN, A. Texture classification: Are filter banks necessary? In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.* June 2003, vol. 2, p. 691–698.

[14] SHU LIAO, CHUNG, A. C. S. Texture Classification by using Advanced Local Binary Patterns and Spatial Distribution of Dominant Patterns. In *Acoustics, Speech and Signal Processing, ICASSP 2007*. 15-20 April 2007, vol. 1, p.I-1221-I-1224.

[15] BAUTISTA, P. A., LAMBINO, M. A. *Co-occurrence Matrices for Wood Texture Classification*. Electronics and Communication Department, College of Engineering MSU-Iligan Institute of Technology, 2001.

[16] PARTIO, M., CRAMARIUC, B., GABBOUJ, M., VISA, A. Rock texture retrieval using gray level co-occurrence matrix. In *NORSIG-2002, 5th Nordic Signal Processing Symposium*. On Board Hurtigruten M/S Trollfjord (Norway), October 4-7, 2002, CD-ROM.

[17] YU-LEN HUANG, RUEY-FENG CHANG Texture features for DCT-coded image retrieval and classification. In *Proceedings of ICASSP '99 Acoustics, Speech, and Signal Processing*, 15-19 Mar 1999, vol. 6, p. 3013-3016.

[18] ORAVEC, M., PAVLOVIČOVÁ, J. Face recognition methods based on principal component analysis and feedforward neural networks. In *Proc. of the International Joint Conference on Neural Networks IJCNN 2004*. Budapest (Hungary), July 25-29 2004, vol. 1, p. 437-442.

[19] VIBHA S. VYAS, PRITI P. REGE Automated texture analysis with Gabor filter. *GVIP Journal*, vol. 6, no. 1, July 2006, p. 35-41.

[20] OJALA, T., PIETIKÄINEN, M. *Texture Classification*. http://homepages.inf.ed.ac.uk/rbf/CVonline/LOCAL_COPIES/OJALA1/texclas.htm

[21] YU-LEN HUANG A Fast method for textural analysis of DCT-based image. *Journal of Information Science and Engineering*, 2005, vol. 21, p. 181-194.

[22] APALOVIČ, L. *Texture Classification*. Thesis, UK Bratislava, 2009 (in Slovak).

[23] POLEC, J., POHANČENÍK, M., ONDRUŠOVÁ, S., KOTULIAKOVÁ, K., KARLUBÍKOVÁ, T. Error concealment for classified texture images. In *EUROCON 2009: International IEEE Conference devoted to the 150-Anniversary of Alexander S. Popov*. Saint Petersburg (Russia), 2009, p. 1348-1353.

[24] BLUNSDEN, S. *Texture Classification using Non-Parametric Markov Random Fields*. Master thesis, University of Edinburgh, School of Informatics, 2004.

[25] POLEC, J., KARLUBÍKOVÁ, T. Discrete orthogonal transform for gappy image extrapolation. In *Proceedings of Computer Vision and Graphics International Conference, ICCVG 2004*. Warsaw (Poland), September 2004, Proceedings, Series: Computational Imaging and Vision, vol. 32, Springer, 2005, pp. 222-227.

[26] KELEŠI, M., MOKOŠ, M., ORAVEC, M., PAVLOVIČOVÁ, J. Error Concealment in Block Coded Images. In *Proceedings of RTT 2005. 6th International Conference Research in Telecommunication Technology*. Hradec nad Moravicí (Czech Republic), 12- 14 Sep. 2005, VŠB-Technická univerzita Ostrava, 2005, CD-ROM.

[27] SHUIMING YE, GUIJIN WANG, XINGGANG LIN Feature based adaptive error concealment for image transmission over wireless channel. In *Proceedings of SPIE Electronic Imaging*, 2003, vol. 5022, p.820-830

About Authors ...

Želmíra TÓTHOVÁ was born in 1977 in Šaľa, Slovak Republic. She received the Engineer degree in telecommunication engineering from the Faculty of Electrical and Information Technology, Slovak University of Technology in 2001. She is PhD student at the same university and faculty. Her research interests include image coding, interpolation and channel modeling.

Jaroslav POLEC was born in 1964 in Trstena, Slovak Republic. He received the Engineer and PhD. degrees in telecommunication engineering from the Slovak University of Technology in 1987 and 1994, respectively. Since 1997 he has been associate professor and since 2007 professor at the Slovak University of Technology and since 1998 at the Department of Applied Informatics of Comenius University. His research interests include Automatic-Repeat-Request (ARQ), channel modeling, image coding, interpolation and filtering.