Dynamic Textures Modelling with Temporal Mixing Coefficients Approximation

Michal Havlíček*

4th year of PGS, email: havlimi2@utia.cas.cz
Department of Mathematics
Faculty of Nuclear Sciences and Physical Engineering, CTU in Prague

advisor: Michal Haindl, Pattern Recognition Department, Institute of Information Theory and Automation, ASCR

Abstract. Appearance of many real world materials is not static but changes in time. In case of spatially and temporally homogeneous changes the material can be represented by means of dynamic texture. Dynamic texture modelling is a challenging problem. In this article we present possible solution based on eigen analysis of input data and subsequent processing and modelling of temporal interpolation eigen coefficients using a combination of piecewise linear approximation and normal distribution sampling. The proposed method shows good performance, enables compress significantly the original data and extremely fast synthesis of arbitrarily long extension of the original texture.

Keywords: Dynamic texture, texture analysis, texture synthesis, data compression, computer graphics

Abstrakt. Vzhled mnoha skutečných materiálů není statický, ale mění se v čase. V případě prostorově a časově homogenních změn může být materiál reprezentován pomocí dynamické textury. Modelování dynamických textur představuje složitý problém. V tomto článku uvádíme možné řešení založené na vlastní analýze vstupních dat a následném zpracování a modelování časových interpolačních vlastních koeficientů pomocí kombinace po částech lineární aproximace a vzorkování z normálního rozdělení. Navržená metoda dosahuje dobrých výsledků, umožňuje výraznou kompresi původních dat a velmi rychlou syntézu libovolně dlouhého rozšíření původní textury.

Klíčová slova: Dynamická textura, analýza textur, syntéza textur, komprese dat, počítačová grafika

1 Introduction

Dynamic textures (DT) can be understood as spatially repetitive motion patterns exhibiting homogenous temporal properties. Good examples might be smoke, fire or liquids. Also waving trees or straws or some moving mechanical objects can be considered as dynamic textures. A sequence of images which are called frames is a basic representation of DT. Original data are always represented by finite length sequence. This property may limit the use of DTs in virtual reality systems so temporally unconstrained modelling of DT is an interesting problem in research such as computer vision, pattern recognition and computer graphics.

*Pattern Recognition Department, Institute of Information Theory and Automation, ASCR.

Already published works dealing with DTs can be divided according to the application to: recognition, representation and synthesis [1]. The DT synthesis is apparently the most difficult task and there are only few papers on this topic available [2]. For example: spatio temporal causal auto regressive model [7], auto regressive moving average model applied on responses of dimensionality reduction filter based on singular value decomposition [6], generative mono spectral DT model based on moving object structure modelling and trajectory modelling by means of dictionary containing Gabor bases for particle elements and Fourier bases for wave elements [8], combination of spatial steerable pyramid and temporal wavelet transformation [3]. All of them are limited by time consuming synthesis algorithm. In addition method [7] requires some high level of temporal homogeneity of the input and method [3] is restricted on monospectral DTs.

Another possibility is utilize so called video editing techniques, developed for general video sequences originally, which can be used for DT synthesis as DT can be considered as a special case of general video sequence. Several examples: video textures generation based on searching for transition points for looping with additional blending and morphing [5], further extended in [4], or tree structured vector quantization [9]. These techniques are also time demanding, but some of them produce very high visual quality results [9].

The contribution of this paper is to propose straightforward colour DT modelling method with low computational demands enabling extremely fast synthesis of arbitrarily long DT sequence and in addition compression of original data. The method is based on combination of input data dimensionality reduction using eigen analysis and modelling of resulted temporal coefficients by means of combination of piece wise linear interpolation and uncorrelated noise sampling. It was inspired by the method described in [2] and represents interesting alternative.

The rest of paper is organized as follows: Section 2 explains input data dimensionality reduction using eigen analysis, Section 3 describes temporal coefficients modelling, Section 4 deals with DT synthesis, Section 5 presents some achieved results and Section 6 summarizes the article with a discussion.

2 Dynamic Texture Eigen Analysis

The first step is so called normalization of analysed DT in which average frame from all frames in the sequence is computed and then this frame is subtracted from each frame in this sequence. Values corresponding to pixels intensities of individual frames from the normalized sequence are arranged into column vectors forming $(n \times t)$ matrix C where n is a number of values equals frame width $\times$ frame height $\times$ number of spectral planes in the frames and t is a number of frames. Then a covariance $(t \times t)$ matrix A is computed as: $A = C^T C$. The matrix A is decomposed using singular value decomposition so that $A = UDU^T$ where U is an orthogonal matrix of eigen vectors and D is a diagonal matrix of corresponding eigen numbers.

Only $k < t$ eigen vectors corresponding to eigen numbers representing the most of the information are saved. The number k can be determined by several techniques. The threshold selecting vectors which are not used may be computed from the values of the eigen numbers. Assuming that the eigen numbers i.e. the elements $D_{(i,i)}$ are ordered by their value then the threshold δ can be computed as for example:

$$\delta = \frac{1}{t}\sum_{i=1}^{t} D_{(i,i)} \quad or$$

$$\delta = D_{(i,i)} \quad where \quad i = argmin_{j\in\{1,\ldots,k-1\}}(|D_{(j,j)} - D_{(j+1,j+1)}|) \ .$$

Only eigen vectors which fulfill that their corresponding eigen number is higher than the treshold δ are saved. The effects of selecting the threshold δ and therefore the number of preserved vectors k and the other possibilities are further discussed in Section 5 and Section 6.

Eigen images $(n \times k)$ matrix I is computed as: $I = CT$, where T is $(t \times k)$ matrix with elements: $T_{(i,j)} = \frac{U_{(i,j)}}{\sqrt{D_{(j,j)}}}$. Finally a matrix of temporal mixing coefficients of individual eigen images I for all frames from the sequence is computed as: $M = I^T C$. The $(k \times t)$ matrix M is a subject of further processing described in following section.

3 Temporal Mixing Coefficients Processing

A threshold α is computed first: $\alpha = \frac{1}{k}\sum_{i=1}^{k}(\sigma_i)$ where

$$\sigma_j = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n-1}(|M_{(j,i)} - M_{(j,i+1)}| - \mu_j)(|M_{(j,i)} - M_{(j,i+1)}| - \mu_j)} \quad ,$$

$$\mu_j = \frac{1}{n-1}\sum_{i=1}^{n-1}(|M_{(j,i)} - M_{(j,i+1)}|) \ .$$

Then the matrix M is processed following manner: if j-th row of M fulfils $\sigma_j > \alpha$ then mean $\hat{\mu_j}$ and dispersion $\hat{\sigma_j}$ of normal distribution from elements of this row are estimated as:

$$\hat{\mu_j} = \frac{1}{t}\sum_{i=1}^{t} M_{(j,i)} \,, \quad \hat{\sigma_j} = \frac{1}{t}\sum_{i=1}^{t}(M_{(j,i)} - \hat{\mu_j})^2 \ .$$

The row which is under $\sigma_j \leq \alpha$ is disjoint into several sub intervals. We denote the set of the indices representing end points of the rows as L . The right edge i_1 of the block is detected by the threshold μ_j applied to $|M_{(j,i_1)} - M_{(j,i_1+1)}|$ so that at least one row $j_0 \in L$ satisfies $|M_{(j_0,i_1)} - M_{(j_0,i_1+1)}| > \mu_{j_0}$. Then values of $M_{(j,i_0)}$ and $M_{(j,i_1)}$ $\forall j \in L$, where i_0 is the left edge of the block, are saved instead of all values in corresponding interval. In addition blocks with less than two elements are not saved at all. The set of all saved blocks will be denoted as B . The division is driven by the row j^* which both fulfils $\sigma_{j^*} \leq \alpha$ and the average value of all elements of this row is the higher than any other such value of the rest of the rows $j \in \hat{k}$ under $\sigma_j \leq \alpha$.

Another possibility is to disjoint rows into the sub intervals with the same length. The length of intervals affects overall dynamics of the synthesized sequence and it appears that each DT need different division to achieve the best result. Although we have not developed any technique to detect this optimal division yet it is apparent that some semi optimal division sufficient enough exists and it was verified by many experiments that this semi optimal length equals to two percents of the total length of the original sequence.

4 Synthesis

The goal of the synthesis is to create certain number of DT frames so that overall visual appearance is close enough to the original. Unfortunately there does not exist any applicable criterion to decide if the synthesized DT is close enough to the original as explained in Section 5.

During the synthesis a matrix $(k \times t^\dagger)$ of temporal mixing coefficients $M^\dagger$, where $t^\dagger$ is a length of the synthesized sequence, in general different from t, is created block wise from the blocks occurring the set B . Element $M^\dagger_{i,j}$ is linearly interpolated if $j \in L$ or sampled from uncorrelated noise with mean $\hat{\mu_j}$ and dispersion $\hat{\sigma_j}$ otherwise. Blocks may be chosen even non deterministically but $|M_{i_1,j} - M_{i_0,j}| < \mu_j$ must hold for all $j \in L$, i_1 is the right edge of previously used block and i_0 is the left edge of the following one.

New DT sequence $C^\dagger$ which is $(n \times t^\dagger)$ matrix can be then computed simply as: $C^\dagger = M^\dagger U$. Final step is addition of the average frame to each frame in the synthesized sequence. Since only matrix operations occur in this step it can be easily performed on contemporary graphics hardware which considerably increases the synthesis speed.

5 Results

We used the dynamic texture data sets from DynTex texture database [1] as a source of test data. Each dynamic texture from this sets is typically represented by a 250 frames long video sequence, that is equivalent to ten second long video. An analysed DT is processed frame by frame. Each frame is 400×300 RGB colour image. As a test DT were chosen: smoke, steam, streaming water, sea waves, river, candle light, close shot of moving escalator, sheet, waving flag, leaves, straws and branches.

Some results can be seen on Figures 1 and 2, showing selected synthesized frames and corresponding frames from original sequence. In this case the deterministic version of the algorithm with fixed length intervals were use to reproduce the sequence.

From the shown results can be seen that although there are some differences between original and synthesized frames the overall dynamic stayed preserved. Unfortunately it is really hard to express this similarity exactly. Robust and reliable similarity comparison between two static textures is still unsolved problem up to now. Moreover, when we switch to the dynamic textures the complexity of comparison between original and synthesized DT sequence increase even more.

In some cases the synthesized DT is visually similar to the original except for less details (for example: river and straws on Figure 1, sea waves on Figure 2), sometimes the moves in synthesized sequence are blurred (for example: waving leaves and sheet on Figure 2). Less detailed appearance is mainly caused by information loss during the dimensionality reduction phase when only about 15% of the original information is saved.

[1] http://www.cwi.nl/projects/dyntex/

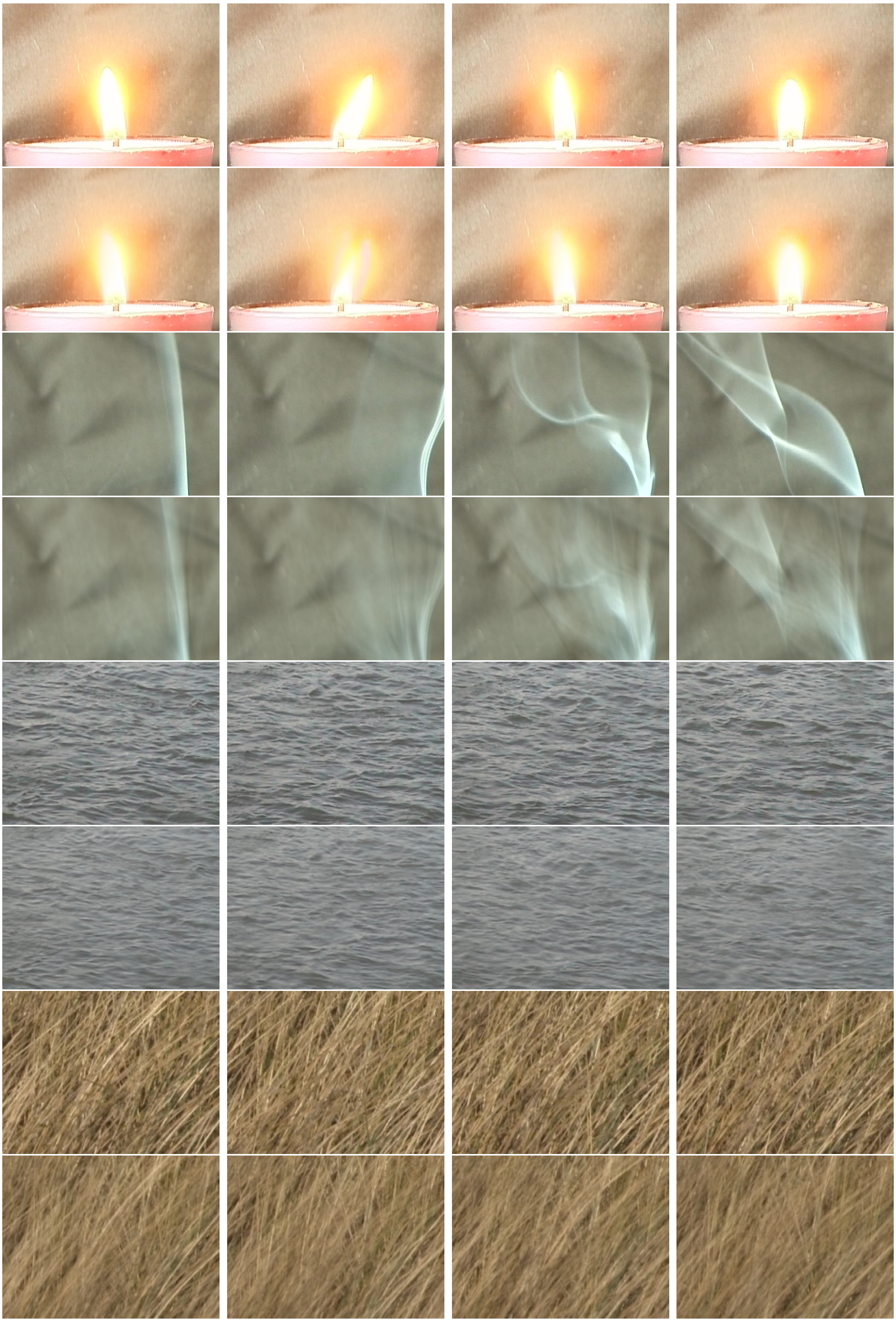

Figure 1: Original frames (odd rows) versus corresponding synthesized ones (even rows), sequences: candle light, smoke, river, straws.

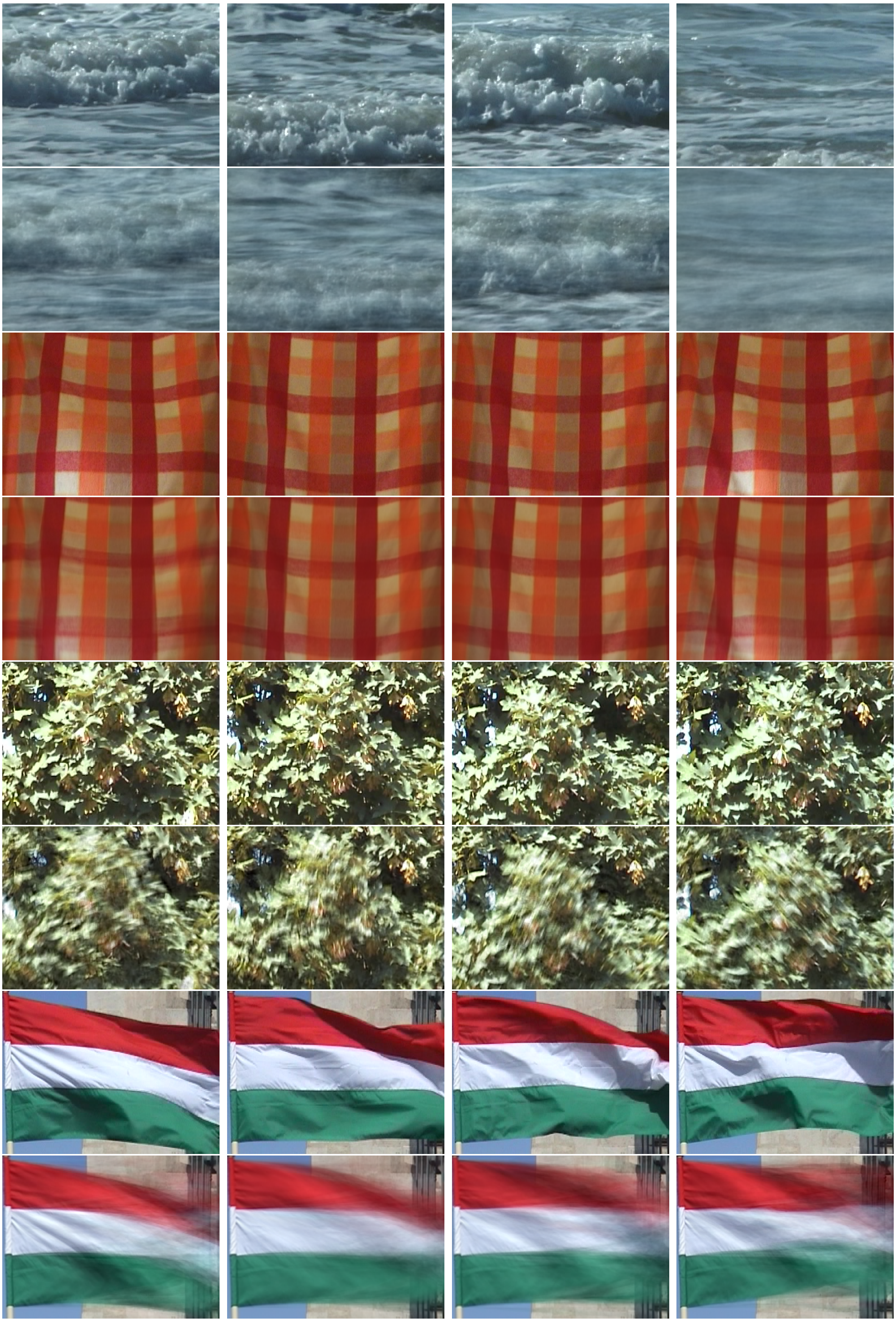

Figure 2: Original frames (odd rows) versus corresponding synthesized ones (even rows), sequences: sea waves, sheet, waving leaves, flag.

Figure 3: The synthesis of several textures (candle light, river, straws and waving leaves) 300th and 400th frames.

The approximation of coefficients is reflected in the blurring. The worst result is the flag sequence synthesis (Figure 2), maybe it is because this is not real DT but rather dynamic scene and this method is limited to DTs.

Main advantage of this method to the solution published in [2], where Causal Auto Regressive (CAR) model is used to process matrix M, is its stability in the synthesis step. Another issue of using CAR model is that the overall dynamics of synthesized sequence decreases with time which is serious problem in case of sequences longer than original one. The general dynamic of the sequence is preserved in time in case of our method as presented on some results on Figure 3 showing selected frames from synthesized sequence longer than original one. The computational demands are identical for both methods.

6 Conclusion and discussion

We presented a novel method for fast synthesis of dynamic multispectral textures in this article. The main part of the approach is based on modelling of temporal coefficients resulted from input data dimensionality reduction step. This solution enables extremely fast synthesis of arbitrary number of multispectral DT frames, which can be even more efficiently performed by contemporary graphical hardware. There are still some unsolved tasks. The detection of optimal number of component which should be saved is still discussed, because this step is essential and affect overall performance and resulting visual quality. The division of temporal matrix is not always the best solution and sometimes the fixed length sub intervals serves as the universal semi optimal solution. On the other we have not developed any method for optimal fixed sub interval length detection yet but many experiments demonstrated that for most DTs 2% of the total length of the sequence is sufficient. Although this method is still under development it represents interesting alternative to the existing approaches.

References

[1] D. Chetverikov, R. Péteri. *A brief survey of dynamic texture description and recognition.* In Proceedings 4th Int. Conf. on Computer Recognition Systems (CORES05), Springer Advances in Soft Computing, (2005), 17–26.

[2] J. Filip, M. Haindl, D. Chetverikov. *Fast Synthesis of Dynamic Colour Textures.* Proceedings of 18th International Conference on Pattern Recognition, IEEE Computer Society Press, **4**, 2006, 25–28.

[3] Z. Joseph, R. El-Yaniv, D. Lischinski, M. Werman. *Texture mixing and texture movie synthesis using statistical learning.* IEEE Transactions on Visualization and Computer Graphics, **7**(2), 2001, 120–135.

[4] V. Kwatra, A. Schödl, I. Essa, A. Bobick. *Graphcut textures: image and video synthesis using graph cuts.* ACM SIGGRAPH 2003, ACM Press, **22**(2), 2003, 277–286.

[5] A. Schödl, R. Szeliski, D. Salesin, I. Essa. ACM SIGGRAPH 2000, ACM Press, 489–498.

[6] S. Soatto, G. Doretto, Y. Wu. *Dynamic textures.* In ICCV, 2001, 439–446.

[7] M. Szummer and R. Picard. *2Temporal texture modeling.* In Proceedings of IEEE International Conference on Image Processing (ICIP 96), 1996.

[8] Y. Wang and S. Zhu. *Analysis and synthesis of textured motion: Particle, wave and cartoon sketch.* IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI), 2004.

[9] L. Wei, M. Levoy. *Fast texture synthesis using treestructured vector quantization.* ACM SIGGRAPH 2000, ACM Press, pages 479–488.