Higher Efficiency of Motion Estimation Methods

Mária GAMCOVÁ, Stanislav MARCHEVSKÝ, Ján GAMEC

Dept. of Electronics and Multimedia Communications, Technical University of Košice, Park Komenského 13, 042 00 Košice, Slovak Republic

maria.gamcova@tuke.sk, stanislav.marchevsky@tuke.sk, jan.gamec@tuke.sk

Abstract. *This paper presents a new motion estimation algorithm to improve the performance of the existing searching algorithms at a relative low computational cost. We try to amend the incorrect and/or inaccurate estimate of motion with higher precision by using adaptive weighted median filtering and its modifications. The median filter is well-known. A more general filter, called the Adaptively Weighted Median Filter (AWM), of which the median filter is a special case, is described. The submitted modifications conditionally use the AWM and full search algorithm (FSA). Simulation results show that the proposed technique can efficiently improve the motion estimation performance.*

Keywords

Motion estimation, Median filtering, Adaptive weighted median.

1. Introduction

The purpose of the motion estimation (*ME*) and compensation is reduction of redundancy caused by interframe correlation of movement objects [1]. However, the estimation and coding of movement vectors should be appropriated to computational costs and bit rates at the perspective high compression systems. That's why relationship between accuracy of movement estimation and simplicity of the description vector fields is very important. Better motion estimation means higher space decorrelation of prediction errors in time area.

The most popular approach is to reduce the number of search locations by using the assumption of unimodal error surface in which the matching error decreases monotonically when the searching location approaches to the global optimum. However, this assumption is not usually satisfied, thus resulting in local optimal solution. Instead of limiting the number of search locations, another interesting technique aims at reducing computation of block matching with pixel subsampling, successive elimination algorithm (*SEA*) [4] or segmentation [12], [13].

The two techniques achieve computation reduction with or without loss of search performance. However, they cannot achieve the better performance than *FSA*. Another direction for fast computation is to exploit the motion correlation between the neighboring blocks in spatial and temporal directions [8]. However, only the correct and fully exploited the spatial and temporal correlations, the improvement of estimation accuracy can then achieve.

For fast search algorithms, the loss of estimation accuracy is due to the simplification process, or the false assumption. On the other hand, the search is performed for integer location, thus the lower precision is also degraded the motion compensation performance.

Since the motion of neighboring blocks in both spatial and temporal are highly correlated. But the motion correlation in spatial neighboring is different from temporal neighboring. In spatial neighboring, the blocks may be partitioned from the same object, thus the neighboring blocks have the motion with similarity. In temporal neighboring, the blocks have the same characteristics of motion, i.e., nearly the same velocity or acceleration.

The further tradeoff between accuracy of *ME* and spatial homogeneity of temporal prediction errors led to the choice of the block matching algorithm (*BMA*) [1] for *ME*, possibly followed by vector field postprocessing [2]. A regularization procedure of motion vectors may be either embedded in the estimation itself [3], or designed for post-processing estimated vectors by exploiting confidence measurements as well.

The *BMA* suffers from another limitation concerning the fidelity of the predicted image: blocking effect is introduced for the lack of coherence of the estimated motion to the actual motion. It is possible to overcome this drawback partially by smoothing the estimated vector field in a further stage. Smoothing can be accomplished, for example, by using weighted median filtering [2], [6], [7], [5] or Kalman filtering [8].

Therefore, in this paper, the main purpose is to present a motion estimation and filtering techniques that give consideration to both estimation performance and computational efficiency.

2. Motion Estimation

For the estimation of the displacement vector (u, v) in a point (x, y) in a frame q, a small matching block centered at the point (x, y) is taken from the frame q and compared with all matching blocks centered at points $(x-u, y-v)$ within a searching area (SA) of the frame q-1. The best match is taken as the presumable displacement vector $\vec{V}_B$ with components (u_B, v_B). Typical (usually used) matching criteria are the mean-square error (MSE) defined as

$$MSE(x,y,u,v) = \\ = \frac{1}{M.N}\sum_{m=1}^{M}\sum_{n=1}^{N}\left[X_q(m,n) - X_{q-1}(m+u, n+v)\right]^2 \tag{1}$$

or the mean-absolute difference (MAD) defined as

$$MAD(x,y,u,v) = \\ = \frac{1}{M.N}\sum_{m=1}^{M}\sum_{n=1}^{N}\left|X_q(m,n) - X_{q-1}(m+u, n+v)\right| \tag{2}$$

where X_q, (X_{q-1}) are picture elements of matching block of frames q, (q-1). The size of the matching block is M by N. Assuming a maximum horizontal or vertical displacement of d_m picture elements ($-d_m \le u,\ v \le d_m$).

The full search (FS) procedure for finding the correlation peak requires an evaluation of MSE or MAD at

$$Q = (2d_m + 1)^2 \tag{3}$$

different horizontal and vertical shifts.

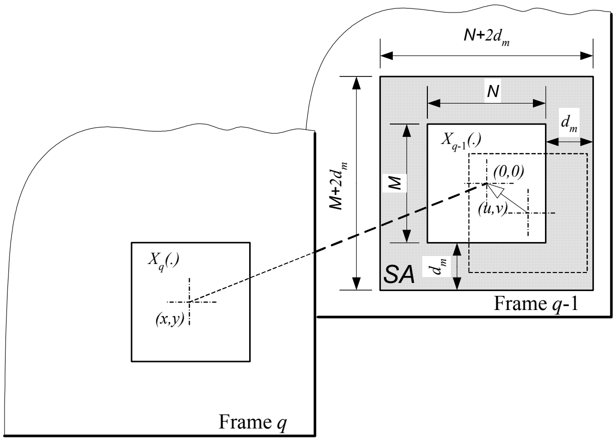


Fig. 1. Positions of subblocks and search area (SA) in the frames q and q-1

In order to reduce computational cost by reducing the high amount of trials, several fast search algorithms for block matching have been developed (*2DLOG*, *TSS*, conjugate direction search methods etc.) [1]. In these methods, the best match of the first step is the starting point of the subsequent step in which the search points are less coarsely spaced.

Another very promising method is block matching with use of the conventional (cross) correlation (CC) function or phase correlation (PC) function [9].

In [10] motion estimation algorithms are proposed, based on the presumption that invertible rapid transform (IRT) consists of the rapid transform (RT), which supplies a shift invariant pattern from the input pattern, and a binary coding process (generating additional data), which records the „phase information“ of the input pattern. Thus additional data are known as a matrix of states (binary matrix) for *1D-IRT* or a system of matrices of states (system of binary matrices) for *2D-IRT*.

The success measure of finding movement vector $\vec{V}_B$ with components (u_B, v_B) considering the block $B(x, y)$ can be the value MAD_B in the position (x, y)

$$MAD_B(x,y,u_B,v_B) = \\ = \frac{1}{M.N}\sum_{m=1}^{M}\sum_{n=1}^{N}\left|X_q(m,n) - X_{q-1}(m+u_B, n+v_B)\right| \tag{4}$$

which one needn't be his minimal value for the fast search algorithms in case of non observance of the monotony criteria condition $MAD(x,\ y,\ u,\ v)$ in SA. In this case $\vec{V}_B$ can be considered a vector corrupted by noise.

To reduce errors in displacement estimate due to false motion detection, a thresholding technique can be adopted. The threshold value T_1 should be set proportional to the variance of the background noise (in our case T_1 was set to 3 or 5 experimentally). If $MAD_B(x,y,0,0)<T_1$, where

$$MAD_B(x,y,0,0) = \\ = \frac{1}{M.N}\sum_{m=1}^{M}\sum_{n=1}^{N}\left|X_q(m,n) - X_{q-1}(m,n)\right| \tag{5}$$

thus the movement vector considering the block B is justified to $\vec{V}_B = (0,0)$ and the searching procedure of the displacement vector is omitted. The binary matrix of search MS can be defined, which elements are

$$MS(k,l) = \begin{cases} 0;\ if\ MAD_B(x,y,0,0) < T_1 \\ 1;\ if\ MAD_B(x,y,0,0) \ge T_1 \end{cases}. \tag{6}$$

For (k, l) it can be written: $x=(l-½)N$, $y=(k-½)M$, $k,l=1,2,\ldots$ The value $MS(k,l) = 1$ indicates in which block the searching procedure was realized.

3. Criteria of Movement Estimation Performance

There are often used subjective criteria or objective ones (i.e. analytically evaluated measures) in the analysis of effectiveness of motion estimation algorithm implementation. The subjective criteria represent subjective regards of larger number of respondents, which categorize picture quality into several levels. The effect of motion estimation can be indicated by the improvement in the signal-to-noise ratio (SNR). We define

$$SNR = 10\log_{10}\frac{\sum_{y=1}^{VS}\sum_{x=1}^{HS}X_q^2(x,y)}{\sum_{y=1}^{VS}\sum_{x=1}^{HS}\left[X_q(x,y) - X^*(x,y)\right]^2} \tag{7}$$

where $X_q(x, y)$ are values of pixels of the frame q, $X^*(x,y)$ are the values of the reconstructed frame or frame q-1, VS is the vertical size of the frame, and HS is the horizontal size of the frame. The frame reconstruction means sub-block settling of the frame q-1 by found movement vectors $\vec{V}_B$.

Another kind of result analysis of motion estimation algorithm implementation is a histogram of displacement vectors drawn in polar form [11].

The movement vector $\vec{V}_B$ can be described in component form (u_B, v_B) or in exponential form $\vec{V}_B = |\vec{V}_B| \exp(i.\varphi_B)$. The value of a histogram

$$H(\alpha_P) = \sum_{\varphi_B \in \langle \alpha_P, \alpha_{P+1})} |\vec{V}_B| \exp(i.\varphi_B);$$
$$\alpha_P \in \{\alpha_1, \alpha_2, \ldots \alpha_S\}, \tag{8}$$

$$H_{Norm}(\alpha_P) = \frac{|H(\alpha_P)|}{\max\{|H(\alpha_P)|\}}, \tag{9}$$

where S is the number of angle divisions 360 degrees. Thus the histogram drawn in polar system well signs the activity in a particular direction $\alpha_P \in \{\alpha_1, \alpha_2, \ldots \alpha_S\}$. An increasing movement activity is indicated if a large object is moved with a small displacement or a small object (several small objects) is moved with a large displacement between consecutive frames.

4. Weighted Median Filtering

The regularity of the estimated vector field, which allows an efficient coding, may be compromised by the constraint of motion vectors (MV) being mathematically matching instead of physically matching. Median filters are suitable for smoothing digital signals/images, as they couple outlier filtering with edge preserving capabilities, thanks to their closure property, by which the filter is constrained to output one of the input samples.

4.1 Median

Let's $Z=\{z_1, z_2, \ldots, z_\Lambda\}$, with $z_\lambda \in \mathbf{R}$, $\lambda = 1, 2, \ldots, \Lambda$. Z is a set of Λ scalars. The fundamental property of sample median z_M states that the average absolute difference from z_λ'is minimized by z_M:

$$\begin{cases} z_M \in Z \\ \sum_{\lambda=1}^{\Lambda} |z_M - z_\lambda| \le \sum_{\lambda=1}^{\Lambda} |z_j - z_\lambda| \quad j = 1,2,\ldots \Lambda \end{cases}. \tag{10}$$

4.2 Weighted Median

The weighted median WM is defined by replicating the λ-th sample within the window as many times as its integer weight w_λ, before finding out the median. Such a definition is extended to the case of real valued weights $W=\{w_1, w_2, \ldots, w_\Lambda\}$, with $w_\lambda \in \mathbf{R}^+$. The definition of WM leads to extending the property stated in (9):

$$\begin{cases} z_{WM} \in Z \\ \sum_{\lambda=1}^{\Lambda} w_\lambda |z_{WM} - z_\lambda| \le \sum_{\lambda=1}^{\Lambda} w_\lambda |z_j - z_\lambda| \\ j = 1,2,\ldots \Lambda \end{cases}. \tag{11}$$

For positive non-integer weights the filtering procedure involves sorting the samples inside the filter window; adding up the corresponding weights from the upper end of the sorted set until the sum just exceeds the half of the total sum of weights, i.e., $\ge 1/2 \sum_{\lambda=1}^{\Lambda} w_\lambda$; the output of the WM filter is the sample corresponding to the last added weight.

As a consequence of (11) the samples having larger weights are more likely candidates to become (weighted) medians. Thus, the weights control the filtering selectivity and the ability to preserve thin details, thereby leading to a greater flexibility of use than for the unweighted case.

The closure property of median filters may be relaxed by defining a componentwise scalar weighted median filter to be applied to MV's. Unfortunately, the weights cannot be split so as to retain their significance as confidences of the related MV components [5].

5. Fast Search Algorithm Modification

5.1 Modification I

If a value $MAD_B(x,y,u_B,v_B)$ is a success criterion of found displacement vector $\vec{V}_B$ with components (u_B, v_B) considering to the block $B(x, y)$, by comparison of this value with the predefined threshold T_2 we can say that finding by a fast search method has been successful or not. If $MAD_B(x,y,u_B,v_B) \cdot MS(k,l) < T_2$, the fast search method is considered to be successful and the found vector is accepted. However, if $MAD_B(x,y,u_B,v_B) \cdot MS(k,l) \ge T_2$, the fast search method is considered to fail and the found vector is substituted by the vector follow-up performance of FS method obtained. FS method guarantees the least value of $MAD_B(x,y,u_B,v_B)$ considering the respective block.

5.2 Modification II

An adaptive weighted median (AWM) filter is suitable for regularization of block MV's. The idea is to adaptively weight each vector within the processing window according to a measure of its confidence. The processing window's size is 3x3 with actual element in the centre. The matrix of weights of neighboring vectors is defined as follows:

$$W = \begin{bmatrix} w(1,1) & w(1,2) & w(1,3) \\ w(2,1) & w(2,2) & w(2,3) \\ w(3,1) & w(3,2) & w(3,3) \end{bmatrix}, \quad (12)$$

$$w(i,j) = \\ = \frac{MAD_B(k,l,u_B,v_B).MS(k-2+i,l-2+j)+\eta}{MAD(k-2+i,l-2+j)+\eta}, \quad (13) \\ i,j \in \{1,2,3\}$$

where the number η prevents dividing by zero (η = 2^{-10}). Thus the weight of actual *MV* is $w(2, 2) = 1$, the weights of neighborhoods depend on the magnitude of their *MAD*'s. Larger weights are associated with those vectors producing lower *MAD* values, thereby roughly following the shape of the block matching function around its global minimum. Filtration process is performed only for vectors with $MS = 1$. The founded displacement vector by fast search method with a high value *MAD* can be replaced in a process of *AWM* filtering by the vector from a set of neighboring vectors which shows the least value *MAD* for corresponding block.

5.3 Modification III

The output value of AWM filter can be some of neighboring vectors considering the actual vector, which provides smaller value *MAD* in its position, however it needn't provide a less value of $MAD_{AWM}(x,y,u_{AWM},v_{AWM})$ in the position of the actual vector. In this modification, after the process of *AWM* filtering in the position with a value $MS(k,l) = 1$ this value of *MV* (u_{new},v_{new}) is accepted, for which

$$\vec{V}_{new} = \arg\left(\min\begin{pmatrix} MAD_B(x,y,u_B,v_B) \\ MAD_{AWM}(x,y,u_{AWM},v_{AWM}) \end{pmatrix}\right). \quad (14)$$

5.4 Modification IV

Modification IV in comparison with modification III differs only in the process of filtering, which is carried out if

$$MAD_B(x,y,u_B,v_B).MS(k,l) > T_2. \quad (15)$$

This condition minimizes the number of vectors that are submitted to the process of filtering. The computational cost is lower, but the indicator *SNR* is worse, too.

The computational complexity of filtering is quite moderate. The number of vectors subject to the process is extremely small.

6. Simulations and Results

The methods and modifications mentioned above were simulated on a personal computer. The results were found using frames of 256 x 256 pixels quantized uniformly to 8 bits. The simulation was carried out with frames of well-known sequences "Susie"(a picture type portrait with less details) and "Football"(representing frames with high interframe activity). The used block size was 8x8 (16x16) and maximal presupposed displacement was $d_m = 8$. As criteria of movement estimation performance was used *SNR* (7), relative time related to the time needed for finding *MV* with use of *FSA* (Tab. 1).

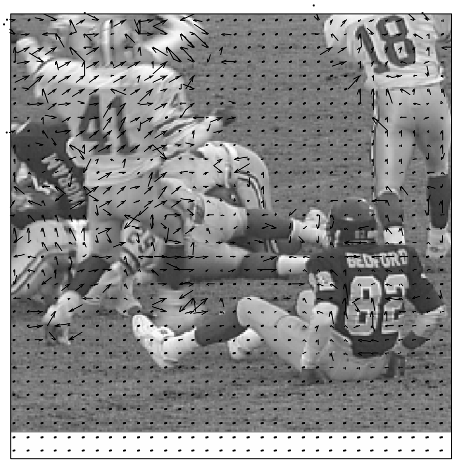
Susie				
	2DLOG	**Median**	**Modification I+III T_1=3, T_2=20**	**FSA**
Relative time	**0,150**	**0,156**	**0,190**	**1,000**
SNR [dB]	**23,90**	**20,06**	**25,00**	**27,45**
Football				
	2DLOG	**Median**	**Modification I+III T_1=5, T_2=40**	**FSA**
Relative time	**0,134**	**0,135**	**0,190**	**1,000**
SNR [dB]	**17,95**	**15,10**	**18,75**	**20,43**

Tab. 1. The typical results showing the computational cost and efficiency (*SNR*) obtained by different modifications from sequences "Susie" and "Football".

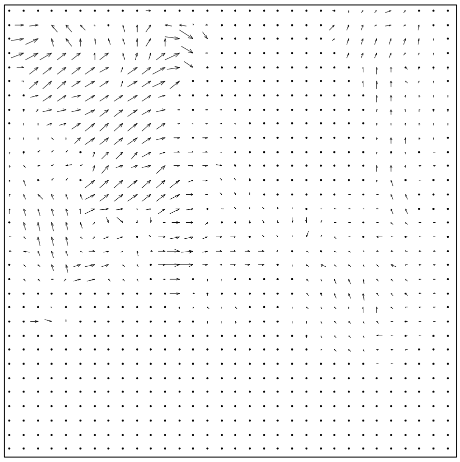
Obtained vector fields, their differences and histograms in polar system (9) ,($\{\alpha_p\}$=$\{0°, 10°, 20°, ..., 360°\}$) can serve for a visual approval of the influence of considering method/modifications (Fig. 2, 3). 2DLOG algorithm was used as a fast search procedure. MATLAB was used in the process of simulation.

7. Conclusion

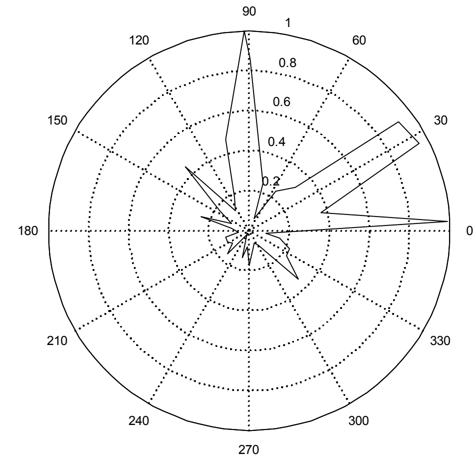
Introduced results indicate that modifications of algorithms can bring small increasing of computational time (with comparison to 2DLOG) but sign an improvement of *SNR* and decreasing computational time with comparison to *FSA*. The more realistic representation of motion in frame after application filtering procedures and their modifications can be seen from vector fields and polar plots. The proposed experiment results indicate good results in terms of computation cost, speed, and motion estimation accuracy.



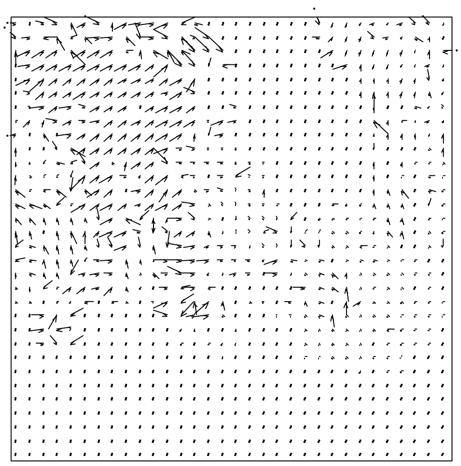
a)



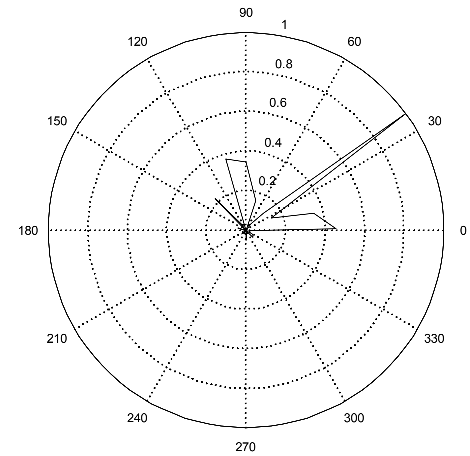
b)



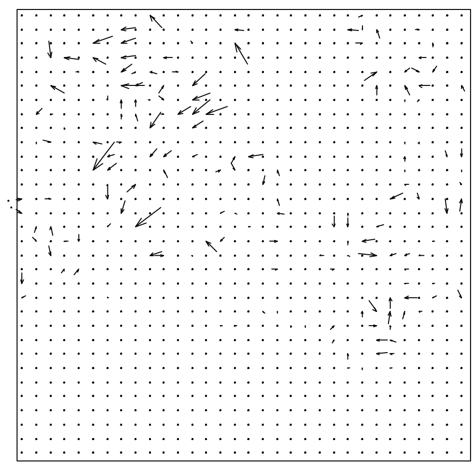
e)



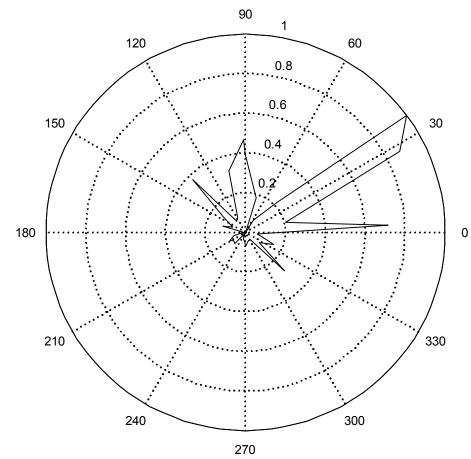
c)



f)

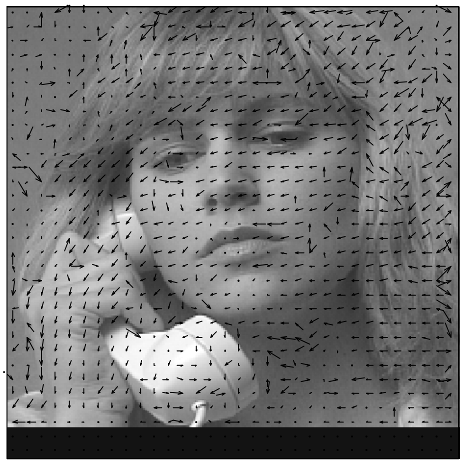


d)

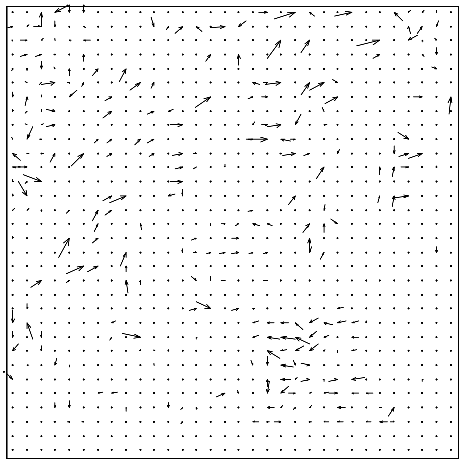


g)

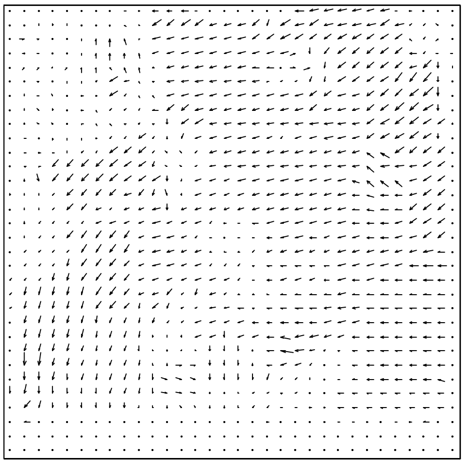
Fig.2. Motion estimation with use of the *2DLOG* method and modifications with block size 8x8, d_m=8:
a) the frame from the image sequence "Football" with motion vectors,
b) the vector field with use of median filtering,
c) the vector field with use of modifications I+III,
d) the vector field of differences between vectors 2DLOG method and vectors of modifications I+III,
e) the polar diagram of vectors 2DLOG method,
f) the polar diagram of vectors after median filtering,
g) the polar diagram of vectors with use of modifications I+III.



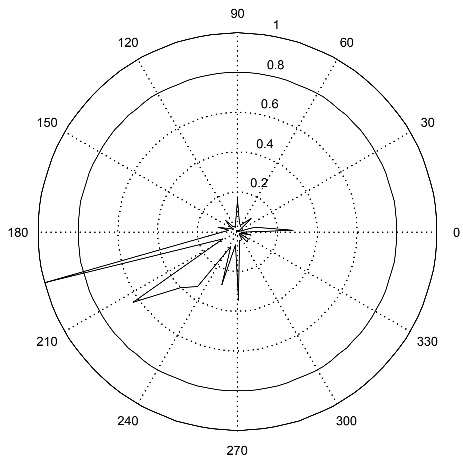
a)



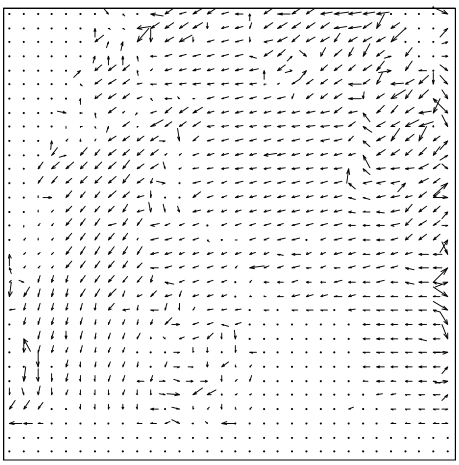
d)



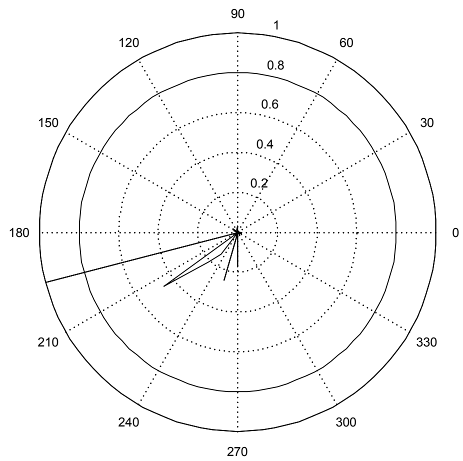
b)



e)

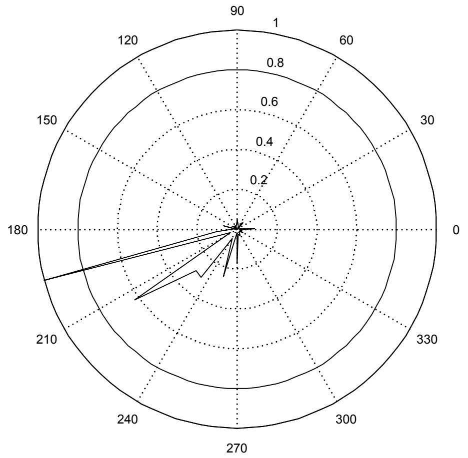


c)



f)

Fig.3. Motion estimation with use of the *2DLOG* method and modifications with block size 8x8, d_m=8:
a) the frame from the image sequence "Susie" with motion vectors,
b) the vector field with use of median filtering,
c) the vector field with use of modifications I+III,
d) the vector field of differences between vectors 2DLOG method and vectors of modifications I+III,
e) the polar diagram of vectors 2DLOG method,
f) the polar diagram of vectors after median filtering,
g) the polar diagram of vectors with use of modifications I+III (please, see the next page).



g)

References

[1] MUSMANN, H. J., PYRSH, P., GRALLERT, H. J. Advances in Picture Coding. In *Proceedings of the IEEE*, 1985, vol. 73, no. 4, p. 523–548.

[2] KOIVUNEN, T., NIEMINEN, A. Motion Field Restoration using Vector Median Filtering on High Definition Television Sequences. In *Visual Communications and Image Processing* ’90, 1990, vol. 1360, p. 736–742.

[3] BARTOLINI. F., PIVA, A. Median Based Relaxation of Smoothness Constraints in Optic Flow Computation. *Pattern Recognition Lett.*, 1997, vol. 18, no. 7, p. 649–655.

[4] LI, W. SALARI, E. Successive Elimination Algorithm for Motion Estimation. *IEEE Trans. Image Processing*, 1995, vol. 4, p.105-107.

[5] ALPARONE, L,. SANTURRI L. Weighted Median Refinement of Motion Vectors in H.263-Based Video Coding. In *IEEE Packet Video Workshop*. Forte Village, Cagliari (Italy), May 2000.

[6] ALPARONE, L., BARNI, M,. BARTOLINI, F., CAPPELINI, V. Adaptively Weighted Vector-Median Filters for Motion Fields Smooting. In *Proceeding of IEEE International Conference on Acoustics Speech and Signal Processing ICASSP”96*. Atlanta (Georgia, U.S.A.), 1996, p.2267-2270.

[7] ALPARONE L,. BARNI, M., BARTOLINI, F., CALDELLI, R. Regularization of Optic Flow Estimates By Means of Weighted Vector Median Filtering. *IEEE Trans. Image Processing*, 1999, vol.8, no. 10, p.1462-1467.

[8] KUO, C. M., HSIEH, C. H., LI, H. C., - LU, P. C. Motion Estimation Algorithm with Kalman Filter. *Electronics Letters*, 1994, vol.30, no.15, p.1204 - 6.

[9] GŐTZE, M. Generation of Motion Vector Fields for Motion Compensated Interpolation of HDTV Signals. *Signal Processing of HDTV,* North-Holland, 1988, p. 383 - 391.

[10] GAMEC, J., TURÁN, J. Inverse Rapid Transform and Motion Analysis. In *Proceedings of Workshop COST 229*, Bayona-Vigo (Spain), 1994.

[11] TURÁN, J., FAZEKAS, K., GAMEC, J., KÖVESI, L. Railway Station Crowd Motion Estimation Using Invertible Rapid Transform. *Image Processing & Communications*, 1997, vol. 3, no. 1-2, p.13-23.

[12] POLEC, J., KARLUBÍKOVÁ, T., PAVLOVIČOVÁ, J. B-th Order Interpolation of Segmented Images Using Shape Independent Sine Transform. In *Proc. of IWSSIP 2000.* Maribor (Slovenia), 2000, p. 85-87.

[13] POLEC, J., PAVLOVIČOVÁ, J. Generalized Interpolation of Segmented Images Using Shape-Independent Orthogonal Transforms. In *Proc. of Int. Conference on Image Processing ICIP 2001.* Thessaloniki (Greece), 2001, p. 888-890.

About Authors...

Mária GAMCOVÁ was born in 1965 in Rožňava, Slovakia. She graduated from the Technical University of Košice with specialization in Radio Electronics, Summa cum laude, in 1989. Since 1989, she has been as an assistant professor of electronics at the Department of Electronics and Multimedia Telecommunications of the Technical University of Košice. Her research comprises linear analogue systems and digital signal processing.

Stanislav MARCHEVSKÝ received the M.Sc. in electrical engineering at the Faculty of Electrical Engineering, Czech Technical University in Prague, in 1976, and Ph.D. degree in radio electronics at the Technical University of Košice in 1985. Currently he is a Professor of Electronics and Multimedia Communication at the Department of the Faculty of Electrical Engineering and Informatics, Technical University of Košice. His teaching interests include switching theory, digital television technology, and satellite communications. His research interests include image nonlinear filtering, neural networks, genetic algorithms, and multiuser detection, space-time communication, and power and bandwidth-efficient multiuser communications.

Ján GAMEC (Ing., Ph.D.), was born in Stuľany, Slovakia in 1960. He graduated from the Technical University of Košice with specialization in Radiotechnics, Summa cum laude in 1985. He reached a Ph.D. degree in radio electronics at the Technical University of Košice, Slovakia, in 1995. Since 1985, he has been an assistant professor in electronics and information technology at the Technical University of Košice. His main area of scientific research is digital image processing.