A New Block Based Motion Estimation with True Region Motion Field

Jozef Huska*, Peter Kulla‡

*Faculty of Electrical Engineering and Information Technology/ Department of Radio and Electronics
Slovak University of Technology in Bratislava, Bratislava, Slovak Republic, e-mail: jozef.huska@stuba.sk

‡Faculty of Electrical Engineering and Information Technology/ Department of Radio and Electronics
Slovak University of Technology in Bratislava, Bratislava, Slovak Republic, e-mail: peter.kulla@stuba.sk

***Abstract*—In this paper a new block based motion estimation algorithm is presented. The algorithm, named as block based motion estimation with true region motion field, is actually based on the successive motion estimation from the highest confidence motion vectors to lowest one. It is utilizing fact that the image pixels corresponding to the same object projection all account similar movement. By estimating motion vector first for the blocks which incorporate at least two different spatial gradient directions (corner blocks) we got reliable estimate for the movement of the other blocks (edge and flat blocks) associated with the same object projection. After than it is only the subject of correct interpolation technique with a refining step to calculate rest of the missing motion vectors. This strategy eliminates chaotic motion vectors and gives motion field more corresponding to a true motion in image sequence, while incorporating small prediction PSNR increase against fast algorithms.**

***Keywords*—motion estimation, true motion field, recursive estimation, multi stage search, search methods.**

I. Introduction

The technique of block based motion estimation (ME) is used in most video coding systems and standards such as MPEG-1/2/4 and H.261/263/264 due to its efficiency. By performing motion estimation and motion compensation we are able to exploit the temporal correlation that exists between frames in a video sequence, and thus achieve high compression. A natural way to exploit redundancy between successive frames is for a current frame in time t to determine predicted frame from the frame in time t-Δt or from the frame in time t+Δt. The motion between frames is described by a motion field of motion vectors.

One of the reasons to look for alternatives for the full search and fast search algorithms (diamond search, 3-step search, ...) is that the resulting motion vectors have poor relation with the true motion of objects, particularly if the signal contains some noise and due to ambiguity of motion estimation (as emphasized in the 2nd part).

Estimating the true motion field does not lead to minimal residue between prediction and original frame but such a relation would be desirable, as true motion vector fields are usually smooth, both spatially and temporally, [1], [2]. That smoothness helps to keep the bit rate low in the typical video codec that applies entropy coding of the motion vector field. Another benefit the true motion vector fields offer is that with the true motion field there is possibility to achieve higher subjective prediction image quality (lower amount of chaotic motion field implies lower amount of wrong displaced image blocks), [3]. On contrary when estimating motion field which leads only to a minimal prediction error is achieved minimal size of compressed residue. The attention must be placed in finding the optimal trade off between minimizing the entropy of the displacement vectors and minimizing the displaced frame difference. The relation between true motion vector field and minimal residue designates the search strategy for estimation of the motion parameters. The strategy should be efficient (complexity) and effective (solution quality).

In this paper is presented a middle way between resulting minimal prediction error and estimating the true motion field. To overcome at once the limitations of motion estimation and make use of true motion field marks our search strategy is based on the successive motion estimation from the highest confidence motion vectors to the lowest one. The algorithm, which is in detail described in the 4th part, uses feature detection algorithm [4] to determine the highest confidence blocks for estimating reference motion vectors for the image regions defined with graph based image segmentation, [5]. The reference motion vectors estimates are used for interpolation of motion vectors for other blocks associated with the same region. The interpolated values of motion vectors are than refined with modified diamond search with the zero position identical to the interpolated motion vector. Motion estimation is done when the motion vectors of all image regions are known. By this approach is guaranteed the smoother motion vector field inside the region as is shown by the simulation results in the 9th part.

For the comparison of there presented new ME algorithm are selected today most used algorithms and as well as the algorithms (selected from the 3rd part) that try to minimize chaotic motion vectors too. Comparison is done with a help of real and synthetic image sequences for which is the correct motion field known. There are used objective characteristics that judge computation time, prediction error and an error of the estimated motion fields against ground truth of the synthetic test sequences.

As can be seen in the last part of this paper our ME algorithm achieves better results as the concurrence especially in the exactness of estimated motion fields in regard to correct motion fields.

II. AMBIGUITY IN MOTION ESTIMATION YIELDING FROM OPTICAL FLOW EQUATION

The luminance variation in a video sequence is represented by I(x,y,t). An image point (x,y) at time t is moved to $(x+d_x, y+d_y)$ at time $t+d_t$. Under the constant intensity assumption (image point has the same intensity along the trajectory of movement) it can be written

$$I\left(x+d_x, y+d_y, t+d_t\right) = I\left(x, y, t\right). \tag{1}$$

Using Taylor's expansion, when d_x, d_y, d_t are small, and combining the result with (1) yields

$$\frac{\partial I}{\partial x} d_x + \frac{\partial I}{\partial y} d_y + \frac{\partial I}{\partial t} d_t = 0 \tag{2}$$

in terms of the motion vector (d_x,d_y). Dividing both sides by d_t it results in

$$\frac{\partial I}{\partial x} v_x + \frac{\partial I}{\partial y} v_y + \frac{\partial I}{\partial t} = 0, \tag{3}$$

where (v_x,v_y) represent the components of velocity vector **v**. The above equation is known as the optical flow equation, [6].

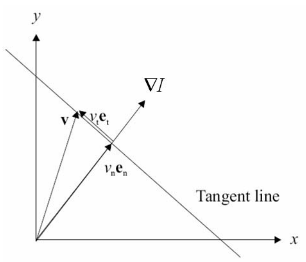


Fig. 1. Decomposition of motion **v** into normal ($v_n\mathbf{e}_n$) and tangent ($v_t\mathbf{e}_t$) components [6].

As shown in Fig. 1., the flow vector **v** at any points x, y can be decomposed into two orthogonal components as

$$\mathbf{v} = v_n \mathbf{e}_n + v_t \mathbf{e}_t, \tag{4}$$

where $\mathbf{e}_n$ is the direction vector of the image gradient

$$\nabla I = \left[\frac{\partial I}{\partial x}, \frac{\partial I}{\partial y}\right], \tag{5}$$

to be called the normal direction, and $\mathbf{e}_t$ is orthogonal to $\mathbf{e}_n$, to be called the tangent direction.

Three consequences as in [6] from (3) are:

*1) Underdetermined compon*ent v_t: there is only one equation for two unknowns (v_x and v_y, or v_n and v_t). To deal with the underconstrained nature of (3) it is convenient apply the motion measurement to a block of pixels [7].

2) Aperture problem: the projection of the motion vector along the normal direction is fixed, with

$$v_n = -\frac{\partial I}{\partial t} / \left\|\nabla I\right\|, \tag{6}$$

where as the projection onto the tangent direction, v_t , is undetermined. Any value of v_t would satisfy the optical flow equation. The word "aperture" refers to the small window over which to apply the constant intensity assumption. The motion can be estimated uniquely only if the aperture contains at least two different gradient directions.

3) Indeterminate flow vector: in regions with constant brightness so that $\left\|\nabla I\right\| = 0$, the flow vector is indeterminate. The estimation of motion is reliable only in regions with brightness variation, i.e., regions with edges or non-flat textures.

III. ANOTHER ALGORITHMS EMPLOYING TRUE MOTION FIELD MARKS

There exist optimization frameworks that provide a formal solution to a problem of finding an optimal trade off between minimizing the entropy of the displacement vectors and minimizing the displaced frame difference, [8]. Such a framework would not increase in a large scale the computation amount of the estimation in comparison with fast algorithms. These frameworks can be categorized in: 1) smoothing the estimated motion vector field, 2) recursive techniques and 3) multi stage methods.

A less complex, but sub-optimal solution, applies smoothing, of the estimated vector field to reduce the bit-rate needed for the coding of the vectors, [6]. Care must be taken that this smoothing does not increase the prediction error too much.

An opposite approach is proposed in [9] where is used spatio-temporal recursive (3DRS) block matching. Using the 3DRS algorithm it leads to consistent, near true motion vector fields through use of recursion. It is known as a high performance motion estimator the best in a blocks based class for applications where the true motion field is required, [10]. In [11] is described improved 3DRS algorithm. The modified recursive search based algorithm examines fewer candidate motion vectors but resulting around 50% improvement in computational load over the conventional 3DRS estimator and simultaneously it provides 8% improvement in the coded bit-rates. Very similar to 3DRS is the predictive diamond search (PDS) algorithm described in [19]. The modification is in used another reference motion vector candidates and refinement stage. Another solution like in [11] and [19] was published in [12]. There are considered the global motion trends of an entire neighbourhood of estimated motion vector by imposing neighbourhood sensitivity into score function. The multi-candidate pre-screening has succeeded in eliminating many wrong motion vectors but there was observed minor tracking errors on points along long edges. The convergence behaviour of the recursive algorithms towards variations in the motion vector field is superior.

Then there are methods which could be named multi-stage that tend to replace wrong vectors estimates by vectors from the blocks with more than one spatial gradient. Algorithm in [13] estimates first the blocks with multiple spatial gradients with robust estimation and blocks with only one gradient are estimated by using a supplementary vector, in particular, the second eigenvector is replaced by the eigenvector of another block from the image area with a common edge. Another similar algorithm is presented in [14], which measures the confidence of the estimation. If the estimation error is large it is used bilinear interpolation from high confidence vector estimates instead. In [3] an initial motion vector is derived from the motion field resulting from spline based registration. Block matching is then carried out to refine that initial estimates. The product is smooth motion field representative of the true motion in the scene.

The algorithms that produce nearly true motion field but are out of this categorization due to large computation demands are gradient based methods such as Lucas-Kanade method, [15]. The gradient methods work well for scenes with small displacement. For estimating larger displacements they are usually implemented with multi-resolution approach, [6].

IV. Used models and estimation criteria

There are some few aspects to consider before development of a new motion estimator for the video coding systems. That premises are:

- the goal is to estimate the motion of image points, i.e., the 2-D motion or apparent motion,
- motion is estimated based on the variations of intensity and color in image points,
- in regard to a projection of 3-D scene objects onto the image plane, the motion of neighboring image points within the object's projection area is very similar, [2].

These facts can be transformed into three important elements which impose the basic parameters of the motion estimation algorithm: models, estimation criteria, and search strategies, [7]. The selection of these elements properties highly designates computational complexity and solution quality. From that sight there are used as simple models and criteria as possible to develop a new motion estimation algorithm which could be used in the existing video coding standards (MPEG-1/2/4, H.261/263/264). That implies using 2-D translational model as spatial motion model and rectangular block of pixels ($R_x x R_y$) as region of support (the set of image points to which a spatial and temporal model applies). That selection returns algorithm from the family of block based motion estimators. As temporal motion model it is used linear trajectory model, constant intensity assumption along a motion trajectory (1) stands for observation model and as the estimation criterion is used a minimization of a function of the following error:

$$\varepsilon(d_x, d_y) = \sum_{x\in R_x} \sum_{y\in R_y} \Big(I(x,y,t) - \\ - I(x+d_x, y+d_y, t+\Delta t) \Big)^2 . \qquad (7)$$

Once models have been identified and incorporated into the estimation criterion the last step is to develop an efficient (complexity) and effective (solution quality) strategy for finding the estimates of motion parameters.

V. Region based multi-stage motion estimation framework for the algorithm

The key part of a new motion estimation algorithm is the search strategy in which the new algorithm differs from the existing. It should offer better solution quality too.

The basic thought of the search strategy we used is to get around the ambiguities in motion estimation (as emphasized in the 2nd part) while keeping the prediction error in an acceptable rate.

The first ambiguity does not bother because of applying motion estimation to a block of pixels.

The second ambiguity, the aperture problem, is worked out by the estimation of motion vector fields only for the blocks including minimal two different spatial gradients directions (so called "corner blocks"). For the blocks including only one spatial gradient (so called "edge blocks"), where occurs the aperture problem, is the motion vector value first interpolated from the corner blocks. The interpolated motion vector estimate is then gently refined. By this way the missing tangential direction to spatial gradient is recovered from the surrounding corner blocks.

The estimation of motion vectors for blocks with minimal intensity change (so called "flat blocks") is similar to the "edge blocks" only with a difference that for interpolation are token corner and edge blocks too. With this procedure is done better estimation of the motion vectors for blocks associated with the third motion estimation ambiguity.

Our search strategy could be characterized as multi-stage, due to estimating the motion vectors in successive steps, from highest confidence estimates (motion vectors for corner blocks) to the lowest one (motion vectors for flat blocks). Used proceeding in the stages is defended by the observation in [14], where described experiments indicate that the interpolated motion fields based on sparse but reliable estimates are more accurate than unreliable dense estimates.

The assumption that the motion of neighboring image points within the object's projection area is very similar (Fig. 2.) is in our algorithm included in a bigger manner than only estimating motion vectors for blocks of image points. Let assume that the object from the scene is projected in 2-D image in such a way, that the object's projection area, so called "region", is larger than one block of pixels. Than it could be assumed that the motion of all blocks within the same region is very similar. By embedding the image regions into proposed motion estimation search strategy it is given that:

- motion vector field inside each region would be smooth,
- interpolation of the motion vectors is done with vectors within the own region, i.e., it ensures different motion vector field between image regions.

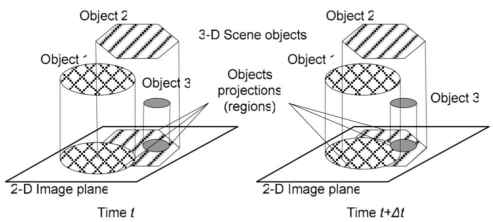


Fig. 2. 3-D Scene objects projection in 2-D image plane.

VI. Detection of the Image Regions, Corner, Edge and Flat Blocks

The selection of the corner blocks is done with a feature detection algorithm [4,14,16,17] which is based on the relationship between eigenvalues λ_1, λ_2 (9) of an image covariance matrix A:

$$A = \sum_{x \in R_x} \sum_{y \in R_y} \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix}, \tag{8}$$

where

$$a_{11} = \left(\frac{\partial I(x,y,t+\Delta t)}{\partial x} \right)^2, \quad a_{12} = \frac{\partial^2 I(x,y,t+\Delta t)}{\partial x \partial y},$$

$$a_{21} = \frac{\partial^2 I(x,y,t+\Delta t)}{\partial y \partial x}, \quad a_{22} = \left(\frac{\partial I(x,y,t+\Delta t)}{\partial y} \right)^2.$$

$$\lambda_{1,2} = \frac{1}{2}\left[\left(a_{11} + a_{22}\right) \pm \sqrt{4a_{12}a_{21} + \left(a_{11} - a_{22}\right)^2} \right] \tag{9}$$

Summarizing the relationship between the matrix A and image block structure, two large eigenvalues of A represent corners. The example is shown in Fig. 3. d).

An image is segmented into regions with the graph based segmentation algorithm described in [5]. It is used for its fast execution time. An important characteristic of used segmentation is its ability to preserve detail in low variability image regions while ignoring detail in high variability regions. The regions after segmentation with projections of scene objects are very similar. The image regions are after the segmentation fitted for image blocks grid, in so far that each region has its: boundary blocks that create the boundary of a region and inner blocks comprehending all blocks between boundaries of a region. The example of the image segmentation with blocks overlay and classification is shown in Fig. 3.

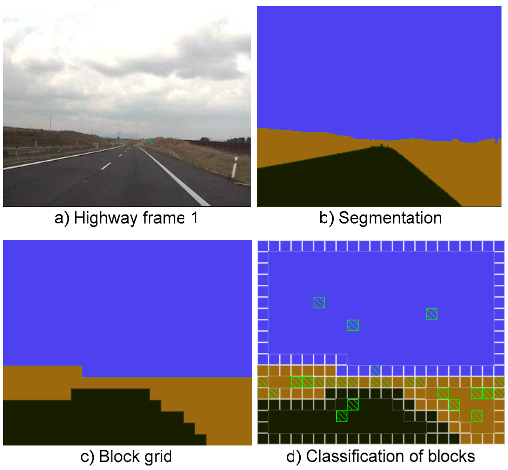


Fig. 3. Example of b) image segmentation, c) fitting a segmented image into block grid and d) blocks classification into boundary blocks (white blocks), flat blocks (without marking), corner blocks (shaded green blocks). Parameters of the segmentation: smooth factor=0.5, segmentation threshold=1000, min block size=1000 pixels, parameters for the feature detection: max difference between λ1, λ2 60%, threshold for min accepted λ1, λ2 9500ppm from max theoretic value.

Resulting classification is used to assign that the boundary blocks are taken as edge blocks and inner blocks as flat blocks. In the case of image block is detected as corner and concurrently as boundary or inner block the corner block flag has a priority.

VII. Motion Vector Field Interpolation and Refinement Stage

Interpolation of the motion vectors for the edge blocks is done with the motion vectors of the corner blocks that were estimated with the diamond search (DS) motion estimator. Before the interpolation are found four nearest reference (corner) blocks $\{RB_i\}_{i=1}^4$ with motion vectors estimates $\{(d_{xi}, d_{yi})\}_{i=1}^4$ from the same region as the edge block. The search is executed in the four directions: north, east, south and west. The interpolated motion vector value $(\tilde{d}_x, \tilde{d}_y)$ is calculated as weighted mean value, [18]:

$$\tilde{d}_x = \frac{\sum_{i=1}^4 w_i d_{xi}}{\sum_{i=1}^4 w_i}, \quad \tilde{d}_y = \frac{\sum_{i=1}^4 w_i d_{yi}}{\sum_{i=1}^4 w_i}, \tag{10}$$

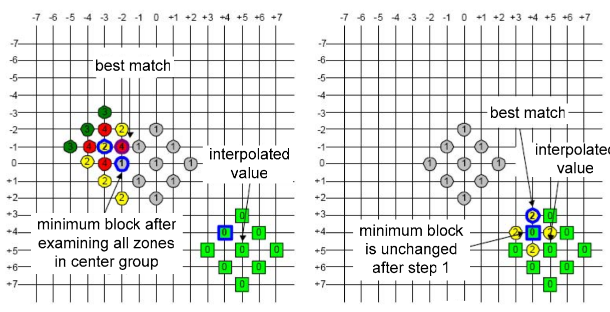
where are used weights

$$\{w_i\}_{i=1}^4 = \left\{ \frac{\min\{|RI_i|\}_{i=1}^4}{|RI_i|} \right\}_{i=1}^4 \tag{11}$$

involving the distance $|RI_i|$ between the reference (corner) and a block for it is currently interpolated motion vector (edge block).

Interpolation of the motion vector for the flat blocks is done with the same weighted mean function. Only difference is in used reference blocks. As a reference blocks are considered nearest blocks from corner and edge blocks located in the same image region.

Predictive diamond search described in [19] was used for the purpose of refinement with small modification. The modification is in used zero position value. Instead of used zero position from the motion vector estimate from candidates like in 3DRS algorithm there is used interpolated motion vector $(\tilde{d}_x, \tilde{d}_y)$ as shown in Fig. 4.



a) zero position at (0,0) b) zero position at $(\tilde{d}_x, \tilde{d}_y)$

Fig. 4. Definition of the modified predictive diamond search algorithm.

VIII. ALGORITHM DESCRIPTION

The algorithm for block based motion estimation with true region motion field (TRMF) includes the results from the all previous sections in a compact form represented with the scheme in Fig. 5. A process of motion vector field estimation can be distributed into stage of preprocessing and the own motion estimation with the proposed search strategy. The stage of preprocessing includes image features detection, image segmentation and a classification of image blocks in the regions.

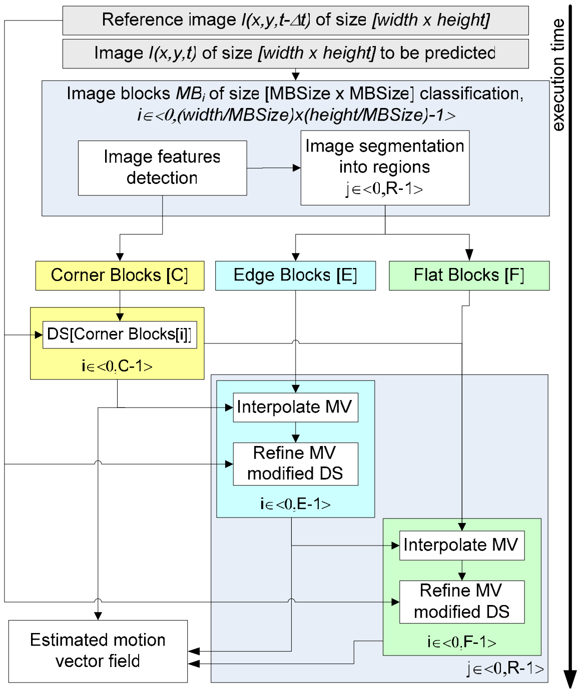


Fig. 5. Algorithm for block based motion estimation with true region motion field (TRMF). (DS-diamond search, MV - motion vector, R-count of regions, C-count of corner blocks, E-count of edge blocks, F-count of flat blocks)

IX. SIMULATION RESULTS

New developed TRMF search algorithm is compared with full search (FS), diamond search (DS), improved 3D recursive search (3DRS) from [11] and predictive diamond search (PDS) described in [19]. Comparison with the FS estimator is incident to confrontation against achievable minimal error between prediction and original image. Results achieved with the DS represent the group of fast search algorithms, 3DRS and PDS algorithms stand for fast block based algorithms that strive to estimate true motion vector field. Our new algorithm invites to the challenge especially 3DRS and PDS algorithms because these three has the same base – DS algorithm. We have examined the performance of our algorithm with the others on real and synthetic sequences for which 2D motion fields were known. There are compared performances of the algorithms in regard to computation demands too. We used relative computation times that are in relative scaling to computation time needed by DS algorithm because we want to know increase/decrease in computation demands against DS.

Set of real test image sequences includes sequences: ”highway” and ”carphone” from [23], ”foreman”, ”mobile & calendar” from [24]. For performance comparison of algorithm applied on real image sequences we used objective quality measurement based on peak signal to noise ratio (PSNR) of the prediction.

Used synthetic image sequences include: ”blocks”, ”grid”, ”office”, ”sphere” ”street” from [20] and ”yosemite” from [21]). The main advantage of synthetic inputs is that the 2D motion fields can be tested in a methodical fashion. In particular, we have access to the true 2D motion field and we can therefore measure estimated motion field similarity to the true motion field. It must be remembered, that such inputs are usually clean signals and therefore this measure of performance should be taken as an optimistic bound on the expected errors with real image sequences. Following [22] we used an angular measure of error which measures errors as angular deviations from the correct space-time orientation. Let motion velocity vector $\mathbf{v_e}$=(dx/dt,dy/dt) be an estimate of a correct velocity $\mathbf{v_c}$. Than the angular error between the nonzero velocities $\mathbf{v_e}$ and $\mathbf{v_c}$ is

$$\psi_e = arc\cos\left(\frac{\mathbf{v_c}\cdot\mathbf{v_e}}{|\mathbf{v_c}|\cdot|\mathbf{v_e}|}\right). \tag{12}$$

Results of algorithms comparison with the real sequences are reported with the summarization table in the Table I. There are the mean prediction PSNR values for 50 frames and relative mean computation times. Graphs of per frame PSNR comparison of prediction images for fifty frames of the real image sequences are in Fig. 6. Value of PSNR at some frame t is calculated from MSE between original frame t and its prediction while frame t-1 is the reference frame. From our results (Table I. and Fig. 6.) for the real sequences it appears that the best prediction quality in terms of prediction image PSNR has undoubtedly FS algorithm. Of course in the terms of estimation speed the FS algorithm is the slowest among the tested algorithms (at average 9.5 times longer computation time than by DS). When comparing our TRMF search algorithm with the DS, TRMF search achieves even higher mean prediction PSNR in the case of “foreman“ (0.01dB) and “mobile & calendar“ (0.27dB) sequences. The prediction PSNR achieved by TRMF search tails away behind DS in maximum 0.14dB in case of “carphone“ sequence. TRMF search has outperformed the PDS in “highway“ sequence and has achieved the same result as PDS in “foreman“ sequence. Algorithm 3DRS is beaten by TRMF search in “highway“, “mobile & calendar“ and “foreman“ sequences. Considering computation demands of there tested algorithms emerges this order of algorithms from fastest to the slowest: 3DRS, DS, PDS, TRMF search, FS. There presented algorithm can be classified as fast algorithm, it has in maximum only 1.21x larger computation demands as DS.

The comparison of our TRMF search algorithm with the FS, DS, PDS and 3DRS algorithm are for the synthetic sequences summed up in the Table II. The experiments show lower angular error of estimated motion fields by TRMF search against DS in all sequences, against PDS in all sequences except “street“ sequence, against 3DRS in all sequences except “office“, “sphere“ sequences, and against FS in “grid“, “office“ sequences. In relation to prediction PSNR our TRMF search has better results as DS in “blocks“, “street“ and “yosemite“, outperforms 3DRS in “blocks“, “street“, “office“, “yosemite“, PDS in “blocks“, “street“, “office“ and “sphere“ sequences.

TABLE I. AVERAGE PSNR VALUES OF PREDICTED IMAGES OF REAL SEQUENCES AND COMPUTATION TIMES RELATED TO DS ALGORITHM (FOR 50 FRAMES AS IN FIG.6., BLOCK SIZE=8X8, SEARCH WINDOW=17X17 PIXELS)

ME algorithm	average values for 50 frames							
	”highway” QCIF		”foreman” QCIF		”carphone” QCIF		”mob. & cal. ” CIF	
	PSNR [dB]	$time_{relDS}$	PSNR [dB]	$time_{relDS}$	PSNR [dB]	$time_{relDS}$	PSNR [dB]	$timere_{lDS}$
FS	36.01	9.71	32.62	9.12	33.47	9.24	23.50	9.77
DS	34.77	1	31.78	1	32.99	1	22.70	1
3DRS	33.68	0.86	30.38	0.81	32.24	0.83	22.56	0.81
PDS	34.63	1.06	31.79	1.03	32.96	1.03	22.98	1.02
TRMF search	34.66	1.21	31.79	1.16	32.85	1.16	22.97	1.17

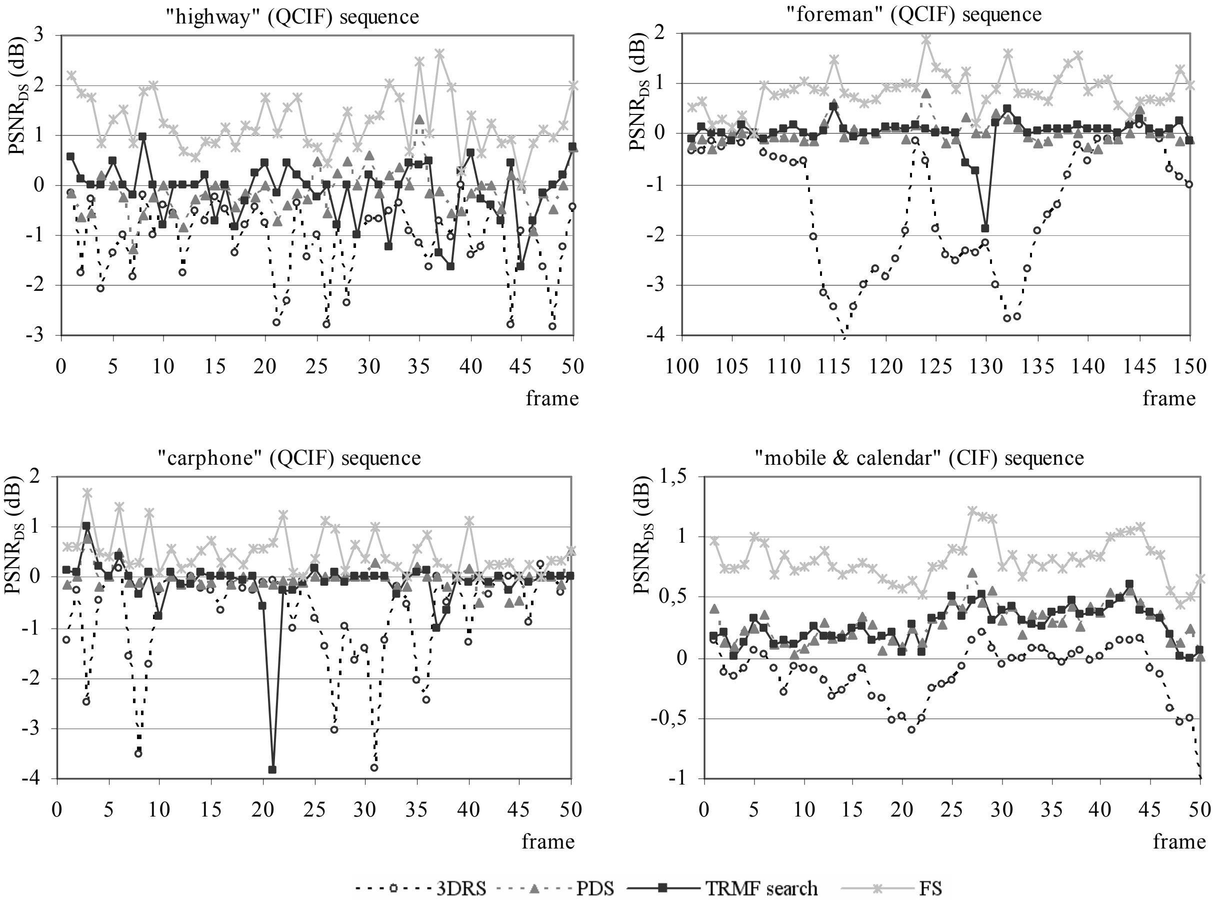


Fig. 6. Prediction image PSNR graphs for the real test sequences. Displayed $PSNR_{DS}$ = PSNR by examined algorithm – PSNR by DS algorithm. (parameters: block size 8x8, search window size = 17x17 pixels, full pixel precision)

TABLE II. COMPARISON OF THE SELECTED BLOCK BASED ALGORITHMS WITH THE TRMF SEARCH ON SYNTHETIC SEQUENCES (BLOCK SIZE=8X8, SEARCH WINDOW=17X17 PIXELS)

ME algorithm	”blocks” (6th frame)		”grid” (2nd frame)		”office” (2nd frame)		”sphere” (2nd frame)		”street” (19th frame)		”yosemite” (2nd frame)	
	PSNR [dB]	mean Ψe [deg]	PSNR [dB]	mean Ψe [deg]	PSNR [dB]	mean Ψe [deg]	PSNR [dB]	mean Ψe [deg]	PSNR [dB]	mean Ψe [deg]	PSNR [dB]	mean Ψe [deg]
FS	24.63	23.84	36.67	67.62	29.68	34.89	45.12	11.21	31.50	4.32	30.73	12.54
DS	21.83	83.77	36.37	67.26	29.44	40.07	43.36	13.46	26.99	12.74	29.56	19.29
3DRS	21.74	91.47	36.37	56.40	26.21	27.05	45.12	11.31	26.26	16.94	28.26	31.27
PDS	21.90	79.55	35.83	62.68	29.10	37.51	42.11	12.93	27.96	7.52	30.14	15.63
TRMF search	22.37	45.85	23.16	29.25	29.38	34.52	43.36	12.92	29.05	8.77	29.94	13.97

X. Conclusions

In this paper a new block based true motion estimation recursive algorithm with multi stage approach was presented. Our new algorithm, named true region motion field (TRMF) search algorithm, uses the same concepts as DS but also considers additional predictive criteria based on recursive and multi stage approach.

The performance of there presented TRMF search was compared with the FS, DS, 3DRS and PDS ME algorithms. The largest competitors of our TRMF search are block based true ME algorithms like 3DRS and PDS. The performance of tested algorithms was measured in relation to prediction quality (PSNR) and angular error between estimated and correct motion field. The performance tests were accomplished on standard real and synthetic image test sequences.

Our presented search strategy significantly improves performance of the DS especially in relation to angular error of estimated motion fields. DS has been outperformed in angular error in all test sequences. Even prediction PSNR was improved in five from ten test sequences. The better output quality is achieved only with the small additional computation demands (in maximum 1.21 longer computation time) against DS. Algorithm 3DRS has been beaten in large scale in both prediction PSNR and angular error. Better results as PDS produced TRMF search without doubt in the case of synthetic sequences. With use of real inputs is the prediction PSNR performance of PDS and TRMF search balanced.

In future work we would like slightly reduce computation demands of our TRMF search and increase output quality. Possible way of doing that would be a modification of image blocks classification or recursive stage of the algorithm (first guess interpolation).

Acknowledgment

This contribution has been supported by the Slovakia Ministry of Education under VEGA Grant No.G-1/3107/06 and VTP 102/2000.

References

[1] A. Verdi, T. Poggio, “Motion Field and Optical Flow: Qualitative Properties,“ *IEEE Transactions on pattern analysis and machine intelligence*, vol. 11, no. 5, May 1989, pp. 490-498.

[2] Y. K. Chen, Y. T. Lin, and S. Y. Kung, “A Feature Tracking Algorithm Using Neighborhood Relaxation with Multi-Candidate Pre-Screening,“ *In Proc. of International Conference on Image Processing*, vol. II., Sept. 1996, pp. 513-516.

[3] F. Dufaux, S. B. Kang, "Combined Spline- and Block-Based Motion Estimation for Video Coding," *In Proc. of 15th International Conference on Pattern Recognition (ICPR'00)*, vol. 3, 2000.

[4] R. Wittebrood, G. de Haan, "Feature Point Selection for Object-Based Motion Estimation on a Programmable Device," *In Proc. of Visual Communications and Image Processing*, vol. 4671, 2002, pp. 687-697.

[5] P. F. Felzenszwalb, D. P. Huttenlocher, "Efficient Graph-Based Image Segmentation," *International Journal of Computer Vision*, vol. 59, no. 2, Sept. 2004.

[6] Y. Wang, J. Ostermann, Y. Q. Zhang, *"Video Processing and Communications,"* Prentice-Hall, 2002, pp. 609.

[7] A. Bovik, *"Handbook of Image and Video Processing,"* Academic Press Series in Communications, Networking, and Multimedia, 2000, pp. 891.

[8] M. E. Al-Mualla, C. N. Canagarajah, D. R. Bull, *"Video Coding for Mobile Communications,"* Academic Press, 2002, pp. 293.

[9] G. de Haan, L. lbaniy, S. Olivieriy, “Noise Robust Recursive Motion Estimation for H.263 based Video Conferencing Systems,” *In Proc. of International Workshop on Multimedia Signal Processing*, Sep. 1999, pp. 345-350.

[10] P. Meuwissen, R. Sethuraman, F. Ernst, “Segment-Based Motion Estimation Using a Block-Based Engine,“ *In Proc. of 13th European Signal Processing Conference (EUSIPCO 2005)*, Sept. 2005.

[11] K. Virk, N. Khan, S. Masud, “Low Complexity Recursive Search Based Motion Estimation Algorithm for Video Coding Applications,“ *In Proc. of 13th European Signal Processing Conference (EUSIPCO 2005)*, Sept. 2005.

[12] Y. K. Chen, *“True Motion Estimation – Theory, Application, and Implementation,“* A dissertation presented to the faculty of Princeton University, Nov. 1998, pp. 235.

[13] Y. Kaneko, Y. Shishikui, Y. Tanaka, “Motion Estimation Based on Eigenvalue Algorithm Matching Local Image Features,“ *Systems and Computers in Japan*, vol. 34, no. 14, 2003, pp. 63-72.

[14] U. Neumann, S. You, "Adaptive Multi-Stage 2D Image Motion Field Estimation," *In Proc. of SPIE Conf. on Applications of Digital Image Processing XXI*, vol. 3460, July 1998, pp. 116-123.

[15] S. Lim, J. Apostolopoulos, A. El Gamal, "Optical Flow Estimation Using High Frame Rate Sequences," *IEEE Transcations on Image processing*, in press.

[16] T. Svoboda, "Kanade-Lucas-Tomasi Tracking (KLT tracker)," http://cmp.felk.cvut.cz/~svoboda/Teaching/CVA4ME/Lectures/Motion/klt.pdf.

[17] C. Harris, and M. Stephens, "A Combined Corner and Edge Detection," *In Proc. of The Fourth Alvey Vision Conference*, 1988, pp. 147-151.

[18] Wikipedia, "Weighted Mean" http://en.wikipedia.org/wiki/Weighted_average.

[19] A. M. Tourapis, G. Shen, M. L. Liou, "A New Predictive Diamond Search Algorithm for Block Based Motion Estimation," *In Proc. of Visual Communications and Image Processing 2000 (VCIP-2000)*, Jun 2000..

[20] B. McCane, K. Novins, D. Crannitch, B. Galvin, “On benchmarking optical flow,” *Computer Vision and Image Understanding*, 2001, p. 126-143.

[21] P. Bayerl, Animations for motion analysis, http://www.informatik.uni-ulm.de/ni/staff/PBayerl/homepage /animations/index.html#yos, online, 27th February 2007.

[22] J.L. Barron, D.J. Fleet, S.S. Beauchemin, "Performance of optical flow techniques," *Computer Vision*, vol. 12, no. 1, Feb. 1994, pp. 43-77.

[23] D. Singer , QuickTime encapsulation of standard video test sequences,http://index.apple.com/~singer/sequences/testseq.html, online, 27th February 2007.

[24] Video Traces Research Group, YUV video sequences, http://trace.eas.asu.edu/yuv/index.html, online 27th February 2007.