Fast Multigrid Pattern Search for Motion Estimation in Hybrid Compression Systems

Nguyen Van Truong and Andrey A. Tropchenko

ITMO University, Saint Petersburg, 197101, Russian Federation,
thientruong.mars@gmail.ru

Abstract. This paper deals with the motion estimation algorithms for the video sequences analysis in hybrid compression standards H.264/AVC and H.265/HEVC. Based on the analysis of the advantages and disadvantages of existing algorithms has been offered a new algorithm, which is called Fast Multigrid Pattern Search (FMPS). This new algorithm includes the Fast Interger-Pel Search FIPS, Adaptive Rood Pattern Search ARPS and hierarchical search MP (Hierarchical search or Mean pyramid). All motion estimation algorithms have been implemented using MICROSOFT VISUAL STUDIO and tested with several video sequences. The criteria for evaluating the algorithms were: speed, peak signal to noise ratio and RD-curves. The proposed method showed a much better performance at a comparable error and deviation (about 4 times faster). The average loss of the RD curve value (PSNR versus bitrate) is up to 3.75% in all. Application of this algorithm in hybrid codecs (H.264/AVC and H.265/HEVC) instead of the standard can significantly reduce compression time. This feature enables to recommend it in telecommunication systems for multimedia data storing, transmission and processing.

Keywords: motion estimation, H.264/AVC, H.265/HEVC, ARPS, FIPS, Fast Multigrid Pattern Search.

1 Introduction

Interframe predictive coding is used to eliminate the large amount of temporal and spatial redundancy that exists in video sequences and helps in compressing them. In conventional predictive coding the difference between the current frame and the predicted frame (based on the previous frame) is coded and transmitted. The better the prediction, the smaller the error and therefore lowered the transmission bit rate. If a scene is still, then a good prediction for a particular pixel in the current frame is the same pixel in the previous frame and the error is zero. However, when there is motion in a sequence, then a pixel on the same part of the moving object is a better prediction for the current pixel [1].

There are many motion estimation algorithms for interframe prediction coding. This work is oriented on algorithms that are called "block matching algorithms" [2-5]. Block matching algorithm is a method of finding block matching in a video sequence for motion estimation. The algorithm includes the division of the current frame into blocks and comparing each of them with a corresponding block in the adjacent frame. It creates a vector which describes the motion of the unit from one place to another. This process is performed for all the frame blocks.

Motion estimation has a fairly large amount of computation and can consume up to 80% of the processing power of the encoder, when a full search (FS) is used. It evaluates all possible candidate blocks within the search window. Outgoing of this disadvantage, looking for other effective algorithms is started.

The Unsymmetrical-cross Multihexagon-grid Search (UMHexagonS) or FIPS was proposed for the fast integer pel motion estimation in H. 264/AVC [6]. Compared to FS, the UMHexagonS algorithm claims that it can reduce 90% of motion estimation time, drop less than 0.05dB PSNR, and maintain the low bit rate, in order to make the initial search point close to the best prediction point. And a novel Center Biased Fractional-pel Search (CBFPS) algorithm or ARPS is proposed for the fast fractional pel motion estimation in [6], which can save 30–50% computation compared with the Full Fractional-pel Search scheme. However, in the UMHexagonS algorithm compared to ARPS and Enhanced Predictive Zonal Search EPZS, the computational complexity is very high, because the search pattern shape has more search candidate. Therefore, in this work we propose fast motion estimation algorithm with name Fast Multigrid Pattern Search FMPS. The proposed method showed that it works about 4 times faster. This is achieved by reducing the number of search pixels on the blocks and the number of coded blocks in the frame. The peak signal to noise ratio in different video sequences shows better and worse results than characteristics of known algorithms (up to 3.75%) so it requires further investigation.

2 Fast Multigrid Pattern Search FMPS

To overcome the disadvantages of existing algorithms, we propose the following algorithm called "Fast Multigrid Pattern Search", which includes the FIPS algorithm for integer pel estimation, ARPS algorithm for fractional pel estimation and Hierarchical Search MP. The whole motion estimation process of FMPS is exemplified in the flowchart in fig. 1:

Component algorithms are discussed in the following sections.

3 Fast Interger-Pel Search FIPS

FIPS is a hybrid method because it includes four steps with different kinds of search pattern [6] (fig. 2):

- Step 1: Initial search point prediction: Spatial median prediction, upper layer prediction, neighboring reference frame prediction, and temporal prediction are used to predict current motion vector (MV) of block;
- Step 2: Asymmetrical cross search. It is followed by an early termination scheme;

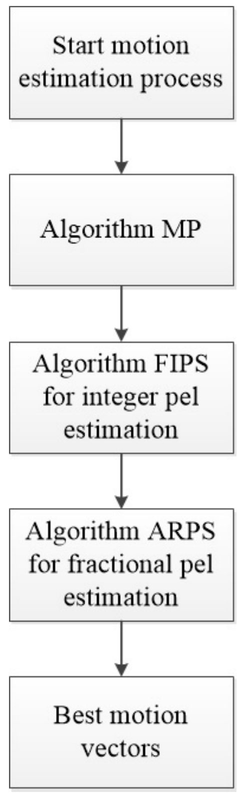

Fig. 1. The flowchart for FMPS algorithm

- Step 3: Uneven multi-hexagon-grid search (step 3-1). Two sub-steps include a square search pattern and a 16 points hexagon search pattern (step 3-2);
- Step 4: Extended hexagon-based search (step 4-1). Two sub-steps include a hexagon search pattern and a diamond search pattern (Small diamond search pattern SDSP) (step 4-2). Early termination scheme is also applied during the search process.

For each step mentioned above, the best search point (which means a point with minimum cost so far) generated by the previous step is used as a search center for the current search step. Initial search point prediction is an important technique introduced by many fast motion estimation algorithms [7,8] setting the search area around the MBD (Minimum Block Distortion) point of the whole search window in order to improve the performance of motion estimation. Median prediction as described in [9] is frequently used in many algorithms and standard. Motion vectors of the collocated block in the previous frame and of the spatially adjacent blocks are also used in [7] as initial search point predictors. four kinds of prediction modes in this proposal:

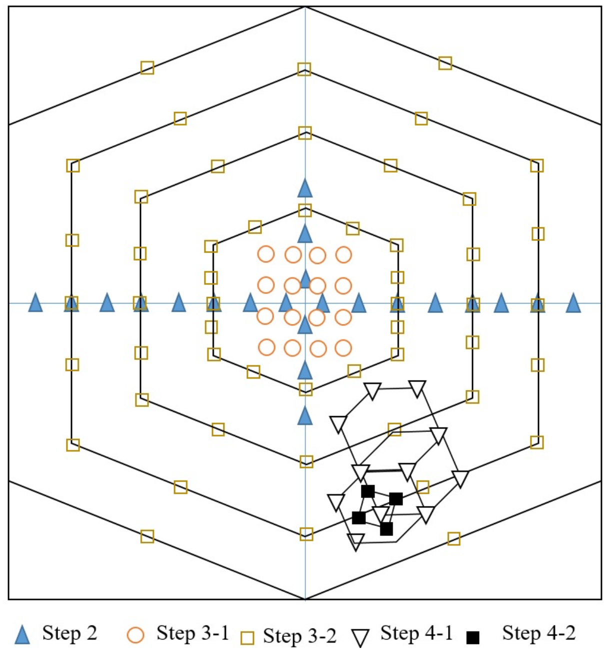

Fig. 2. Search process of FIPS algorithm, W=16

3.1 Spatial median Prediction

As fig. 3 shows, median predictor is used in median prediction of MV, the median value of the adjacent blocks on the left, up, and up-right (or up-left) of the current block is used to predict the motion vector of the current block:

$$MV_{predict} = median(MV_{left}, MV_{up}, MV_{up-right}). \tag{1}$$

Therefore we can predict the SAD (Sum of Absolute Differences) by:

$$SAD_{predict} = min(SAD_{Xpredict}, SAD_{Ypredict}). \tag{2}$$

3.2 Upper layer Prediction

As fig. 4 shows, there are seven inter prediction block modes defined in H.264. 8x8 modes (mode 4, 5, 6, 7) are first searched followed by 16x16 modes (mode 3, 2, 1). Such a strategy is not beneficial in utilizing the motion relationship between different modes, therefore the search order of the modes here is changed according to the size of the block mode, a hierarchically search order from mode 1 to 7 is chosen as our mode search order and the motion vector of the up layer block (for example, mode 5 or 6 is the up layer of mode 7, and mode 4 is the up layer of mode 5 or 6, etc.) is used as one of the prediction candidates of lower layer, just as fig. 5 demonstrates.

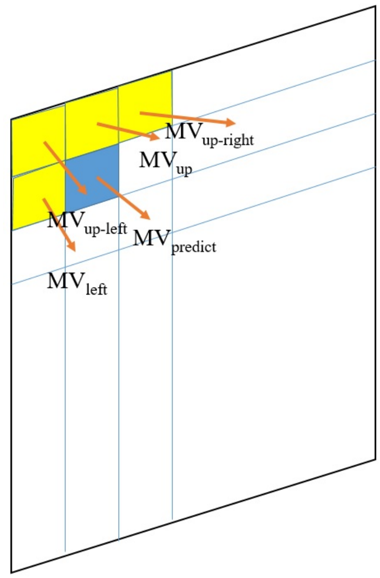

Fig. 3. Spatial median prediction of motion vectors

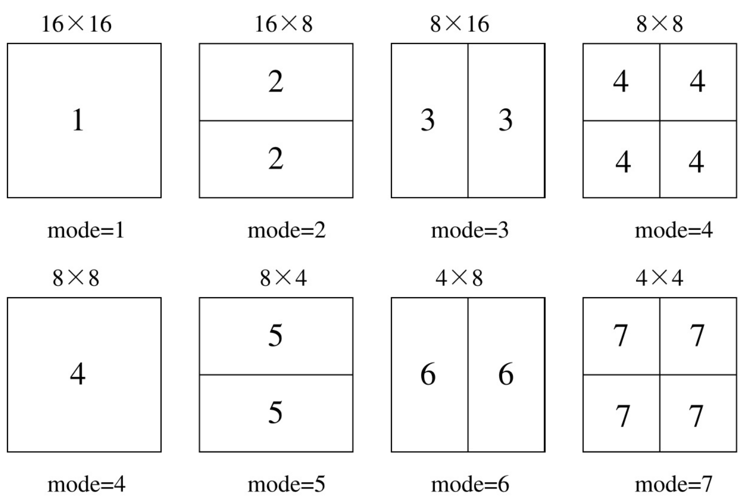

Fig. 4. Seven prediction blocks modes in H.264

$$MV_{predict} = median(MV_{UpLayer}). \tag{3}$$

Therefore we can predict the SAD by:

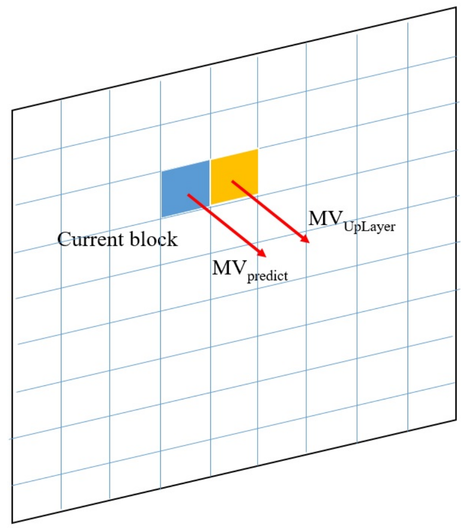

Fig. 5. Spatial upper layer prediction of motion vector

$$SAD_{predict} = \frac{SAD_{UpLayer}}{2}. \quad (4)$$

3.3 Neighboring reference frame Prediction

Multi-reference frames motion compensation is adopted in JVT to increase prediction accuracy and coding efficiency. For the same current block, motion vectors in different reference frames exhibit a strong correlation in our experiment. Therefore current block's motion vector in reference frame tp can be predicted by scaling of current block's motion vector in reference frame tp+1, as fig. 6 shows:

$$MV_{predict} = MV_{neigh} \frac{tc - tp}{tc - tp - 1}. \quad (5)$$

Therefore we can predict the SAD by:

$$SAD_{predict} = SAD_{neigh}. \quad (6)$$

3.4 Temporal Prediction

For natural video sequence, the motion track of a moving object is continuous except scene change occur, therefore there is strong correlation of motion vector in the temporal domain, and then we utilize this property to give an accurate starting search position. In this prediction mode, the motion vector of the corresponding block in the last frame is used as one motion vector candidate, as fig. 7 shows:

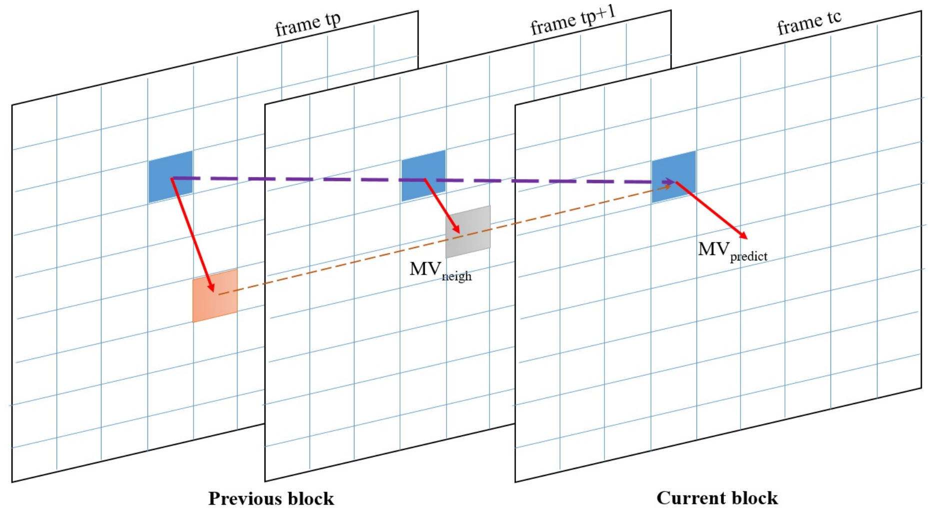

Fig. 6. Temporal Neighboring Ref-frame Prediction of motion vector

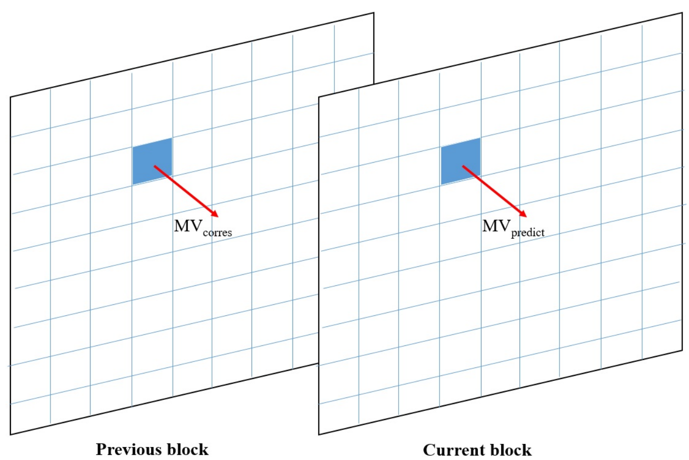

Fig. 7. Temporal Corresponding-block Prediction of motion vector

$$MV_{predict} = MV_{corres}. \tag{7}$$

Therefore we can predict the SAD by:

$$SAD_{predict} = SAD_{corres}. \tag{8}$$

The prediction with the minimum cost among these prediciencies will be chosen as the initial search position of next search step.

4 Adaptive Rood Pattern Search ARPS

The algorithm uses the fact that the total motion of the frame is usually coherent, i.e. if the blocks around the current block moving in a certain direction then there is a high probability that the current block will also have a similar motion vector. This algorithm uses the motion vector of the block to its immediate left to predict its own motion vector [10]. The algorithm summarized as follows (fig. 8):

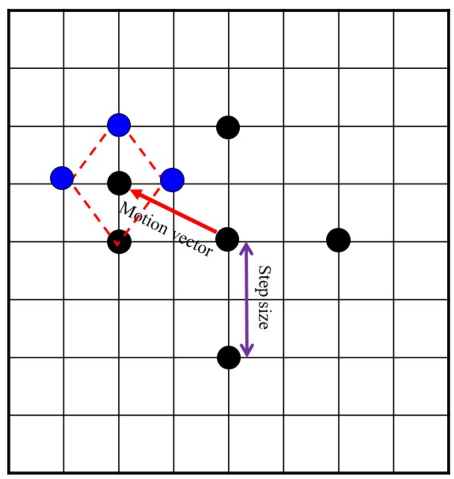

Fig. 8. Example of the ARPS algorithm

- Step 1: Find the predicted motion vector of the block. Set step size as max (|x|, |y|), where (x, y) – coordinates of predicted motion vector. Find points (around the center) are located at a distance of step size from the center. Find the point with the minimum distortion, which then to be the new center.
- Step 2: Perform a search on SDSP around the new center. Repeat SDSP search until point with the minimum distortion is at the center of SDSP.

5 Hierarchical search MP (Mean pyramid)

MP algorithm is described as: At the beginning, to eliminate the effect of noise low-resolution image is obtained by low-pass filter [11, 12]. The scheme of the MP algorithm is shown in fig. 9.

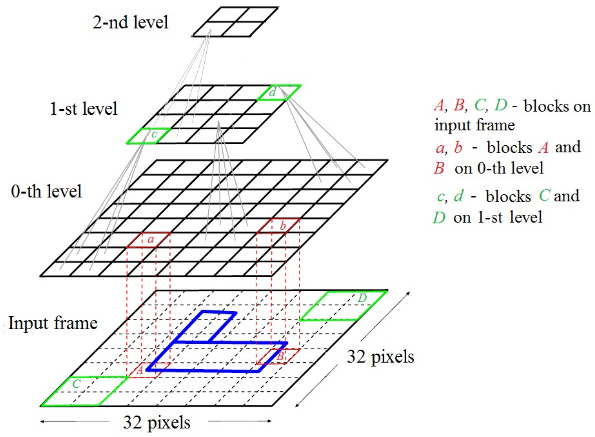

Fig. 9. The scheme of the MP algorithm

6 Experimental results

The proposed algorithm was implemented using Microsoft Visual Studio and tested with several video sequences[1].

In the experiment, we will compare the proposed algorithm with the algorithm FS, FIPS with ARPS[6] and EPZS, which are used in JM 19.0 (H.264/14496-10 AVC Reference Software) by the following criteria: the coding time and the quality of the video sequence (according to PSNR and RD curve). The experiment is carried on Window 10 OS platform with Intel(R) Core(TM) i5-4210U CPU @1.70GHz 2.40 GHz and 6GB RAM.

The coding time is estimated by comparing the average coding time for encoding each video sequence (shown in Table 1). It can be seen that the proposed algorithm reduces the coding time by about 4 times in comparison with the other algorithms.

The quality of the video sequence is estimated by comparing the average PSNR video exponent (Table 2) and the RD curve values (Fig. 10). The results of the research show that the proposed algorithm loses not more than 0.87 to 3.75% of the PSNR ratio in comparison with other algorithms.

In Fig. 10 shows the curves of the dependence of the distortion rate RD of the proposed algorithm and others for the difference video sequences with QP (quantization parameter) equal to 37, 32, 27, 22, vertical axes are PSNR (dB), horizontal axes correspond to bitrate (kbps), and each point on the curves represents the parameter QP. From Fig. 10 that the RD-curves for the considered algorithms (FS, FIPS with ARPS, EPZS and proposed algorithms) are close for each sub-step. This can explain the fact that the proposed change has little impact on both PSNR and bitrate.

[1] Test video sequences `ftp://ftp.tnt.uni-hannover.de/pub/svc/testsequences/`.

Table 1. Average coding time for encoding video sequences

Video Sequences	Average coding time, s			The relative decrease in the time of the proposed algorithm	
	FIPS with ARPS[6]	EPZS	Proposed algorithm	FIPS with ARPS	EPZS
BUS_352x288_15_avc_384,yuv	71.228	66.460	15.794	4.51	4.21
CITY_704x576_30_avc_1024,yuv	239.785	232.819	55.505	4.32	4.19
CREW_704x576_30_avc_1500,yuv	187.483	176.579	46.063	4.07	3.83
FOOTBALL_352x288_15_avc_384,yuv	62.276	59.535	13.085	4.76	4.55
FOREMAN_352x288_30_avc_256,yuv	56.930	54.734	13.208	4.31	4.14
HARBOUR_704x576_30_avc_1500,yuv	276.431	253.878	65.817	4.20	3.86
MOBILE_352x288_30_avc_384,yuv	73.816	71.245	17.961	4.11	3.97
SOCCER_704x576_60_avc_3000,yuv	263.043	245.680	57.433	4.56	4.28

Table 2. Average PSNR for encoding video sequences

Video Sequences	Average PSNR, dB				The relative decrease in the PSNR of the proposed algorithm to the		
	FS	FIPS with ARPS	EPZS	Proposed algorithm	FS	FIPS with ARPS	EPZS
BUS_352x288_15_avc_384,yuv	36.865	36.864	36.864	36.432	1.17	1.17	1.17
CITY_704x576_30_avc_1024,yuv	38.222	38.220	38.221	37.560	1.73	1.72	1.73
CREW_704x576_30_avc_1500,yuv	41.250	41.246	41.249	39.704	3.75	3.73	3.75
FOOTBALL_352x288_15_avc_384,yuv	38.512	38.508	38.511	37.845	1.73	1.72	1.73
FOREMAN_352x288_30_avc_256,yuv	39.153	39.149	39.153	38.031	2.87	2.85	2.86
HARBOUR_704x576_30_avc_1500,yuv	38.680	38.678	38.679	37.383	3.35	3.34	3.35
MOBILE_352x288_30_avc_384,yuv	34.453	34.451	34.452	33.656	2.31	2.30	2.31
SOCCER_704x576_60_avc_3000,yuv	37.915	37.911	37.914	37.584	0.87	0.86	0.87

7 Conclusion

A new algorithm for interframe encoding of the hybrid codecs is proposed, which includes well-known algorithms FIPS, ARPS and MP. The algorithm was tested with several video sequences. Experimental results showed that the proposed algorithm works faster 4 times, while the PSNR coefficient is only 0.86-3.75% below the other algoritms.

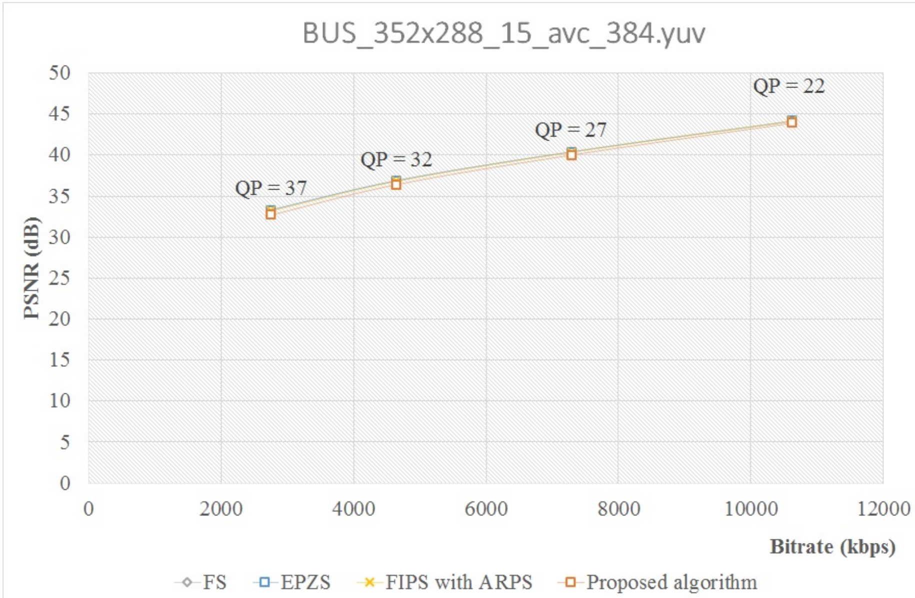

Fig. 10. RD-curves with QP = 37, 32, 27, 22 for video sequence BUS

References

1. *Tauraga, D.* Search algorithms for Block Matching Estimation // Mid-term Project, spring 1998.
2. *Moschetti, F., Kunt, M., Debes, E.* A Statistical Adaptive Block-Matching Motion Estimation // IEEE transactions on circuits and systems for video technology. 2003. Vol. 13, No. 4. P. 417–431.
3. *Babu, D.V., Subramanian, P., Karthikeyan, C.* Performance Analysis of Block Matching Algorithms for Highly Scalable Video Compression // 1-42440731-2006 IEEE. P. 179-181.
4. *Barjatya, A.* Block Matching Algorithms For Motion Estimation // DIP 6620 Final Project Paper in Digital Image Processing, Utah State University. P. 1–6.
5. *Cuevas, E., Zaldívar, D., Pérez-Cisneros, M., Oliva, D.* Block-matching algorithm based on differential evolution for motion estimation // Engineering Applications of Artificial Intelligence. 2013. Vol. 26, No. 1. P. 488–498.
6. *Zhibo , C., Jianfeng, X., Yun, H., Junli, Z.* Fast integer-pel and fractional-pel motion estimation for H.264/AVC // Journal of Visual Communication and Image Representation. 2006, Vol. 17, No. 2, P. 264-290.
7. *Tourapis, H. Y., Tourapis, A., Topiwala, P.* Fast Motion Estimation within the JVT codec // JVT-E023, 5th Meeting: Geneva, Switzerland, 09-17 October, 2002.
8. *Joint Video Team (JVT) of ISO/IEC MPEG & ITU-T VCEG* Joint Video Team (JVT) of ISO/IEC MPEG & ITU-T VCEG // JVT-F100d2.doc, 6th meeting, Awaji, Island, JP, 5-13 December, 2002.
9. *Jain, A. K.* Fundamentals of Digital Image Processing // Englewood Cliffs, NJ: Prentice-Hall, 1989.

10. *Nguyen, V. T., Tropchenko, A. A.* Hierarchical adaptive rood pattern search for motion estimation at video sequence analysis // Scientific and Technical Journal of Information Technologies, Mechanics and Optics. 2016. Vol. 16, No. 3. P. 474–481.
11. *Nguyen, V. T., Tropchenko, A. A.* Methods And Algorithms For Reducing Temporal Redundancy Of Video Data / Sbornik statyei II-oy Mezdunarodnoy nauchno-prakticheskoy konferensyy "Aktualnye problem nauky XXI veka". 2015. Vol. 2. P. 36-41. — Moscow: Cognitio, 2015.
12. *Tropchenko, A. A., Tropchenko, A. U., Nguyen, V.T.* Research of Block-Based Motion Estimation Methods for Video Compression // TEM Jornal - 2016. Vol. 5, No. 3, P. 187-194.