Motion estimation using variable size block matching with cross square search pattern

G. Senbagavalli¹ · R. Manjunath¹

Received: 28 February 2020 / Accepted: 23 July 2020 / Published online: 31 July 2020
© Springer Nature Switzerland AG 2020

Abstract

In recent decades, motion estimation is a major concern in video coding applications. Among all existing techniques, block matching is an emerging technique in motion estimation, because of its simplicity and effectiveness for both hardware and software implementations. In addition, the block matching technique helps in choosing the motion vectors for every macro block instead of utilizing a motion vector for every pixel. In this research article, Variable Size Block Matching (VSBM) with cross square search pattern is proposed for motion estimation in the videos. Compared to other block matching methods, the VSBM results in a low motion compensation error and also significantly speed up the motion estimation process. For reducing the illumination change effect in the video frame, cross square search pattern is developed that prevents macro blocks from over-splitting. In this work, the proposed algorithm performance is analysed by using some real time videos like container, Akiyo, Stefan, kristen-sara, foreman, Johnny, old-town-cross, Carphone, football and crowd-run. From the analysis, the proposed algorithm results in maximum of 14.9 dB improvement in peak signal to noise ratio value as compared to existing algorithms; dynamic pattern (square, diamond and hexagon) and harmony search with block matching.

Keywords Cross square search pattern · Motion estimation in video · Sum of squared distance · Variable size block matching

1 Introduction

In recent periods, multimedia devices have been developed dramatically like mobile phones, tablets, camera, webcam, etc. However, the videos captured utilizing these multimedia devices are easily exposed to the flux problem, due to hand trembling and absence of tools for handling the camera path like steadicam and camera dollies [1, 2]. Under these conditions, it is very hard to understand and watch scenes also it reduce the overall video quality. The concern of unstable video not only arises in mobile platforms and also in the mounted cameras due to resonance of mounting structure and wind speed [3]. In order to address these difficulties, video stabilization procedure

is accomplished for smoothing and enhancing the quality of video [4]. The enhanced video sequence improves the performance in higher image analysis such as visual surveillance, object recognition, object detection, etc.[5, 6]. Generally, the video stabilization contains three stages; motion estimation, compensation and composition. In the motion estimation stage, the camera motion between two succeeding frames are estimated and the resultant frame is passed to the residual stages. In video stabilization, motion estimation is the necessary stage, because the residual two stages depend on the performance of this stage [7, 8].

In video stabilization, recently numerous techniques are developed by the researchers in camera motion estimation

✉ G. Senbagavalli, Shenbab4u@yahoo.com; Shenbab4u@gmail.com; R. Manjunath, manju_r_99@yahoo.com | ¹Department of Electronics and Communication Engineering, AMC Engineering College, Bannerghatta Road, Bangalore, India.

such as clustering [9], pixel wise, and block based motion estimation [10]. In that, block based motion estimation is a popular technique, which is adapted in several video coding standards. In this technique, a video frame is partitioned into several pixel blocks that is named as macro block and then the best match is found in the current frame by searching a particular area in the reference frame on the basis of predefined condition [11, 12]. However, the movement between the present block and its best match is named as motion vector. To manage temporal redundancy, more memory space is required for all motion vectors that highly increases the system complexity. In this research paper, a new cross square search pattern was proposed along with variable block size for achieving better search count, even when the video contains dynamic scenes and more foreground objects. Related to fixed block size, the variable block size lessens the overall complexity by finding whether the fitness value (Sum of Squared Distance (SSD)) of a specific search location was to be estimated or not. Respectively, the cross square search pattern extends the search area and allows more reference frames with fewer computations. So, it was easy to identify the optimal matched block near the center pixel, due to the sparse nature of cross square. Finally, the proposed algorithm performance was related with a few benchmark algorithms in light of Peak Signal to Noise Ratio (PSNR), and Structural Similarity Index (SSIM) and average search count.

This research paper is prearranged as follows; A few existing research papers are surveyed in Sect. 2. The proposed algorithm for motion estimation is significantly detailed in Sect. 3. The experimental consequence of the proposed algorithm is explained in Sect. 4. At last, the conclusion is given in Sect. 5.

2 Literature survey

A. Betka et al. [13] developed a new block matching approach on the basis of Stochastic Fractal Search (SFS) for motion estimation. The developed SFS is a metaheuristic approach, which was utilized to resolve the optimization concerns in limited time. In this literature paper, the developed approach comprises two major phases. In the initial phase, the motion vectors were computed using multi-population model of SFS. In the next phase, a modified fitness approximation methodology was applied for reducing the computational time. In this research, the developed approach was compared with existing block matching approaches for validating the efficiency of proposed approach. In the experimental section, the developed approach outperforms the prior approaches in light of PSNR and Number of Search Point (NSP). In the high

quality videos, the developed block matching approach delivers motion fields with incoherence at artificial locations.

L. Lin et al. [14] developed a new Fast Predictive Search (FPS) algorithm for motion classification on the basis of diamond search algorithm. In the developed algorithm, the initial search point was set near to the optimized search point that helps in identifying the start search point and also avoids inefficient global search process. In FPS, an adaptive search pattern was used for reducing search points and retaining image quality performance. The FPS algorithm lowers the search points in an efficient and precise way by combining motion type classification prediction and stationary state judgment by adopting the early search termination criteria. The experimental consequence shows that the developed algorithm reduces 60% of search points compared to the existing algorithms. Still, the FPS algorithm has some misjudgement chances that leads to PSNR degradation.

M. Wu et al. [15] developed an improved k-means clustering algorithm for global motion estimation. The developed algorithm significantly eliminates the local motion vectors, due to unstable or shaking environment. In this study, speeded up robust feature descriptor was used for matching the feature points by comparing with the existing methodologies performance. In two dimensional feature space, the global motion vectors were calculated by using homography transformation. The detected global motion vectors were used for stabilizing the next frames. The experimental outcome shows that the developed clustering algorithm obtains better video security performance related to the traditional methodologies in light of PSNR and SSIM. In low resolution videos, the developed clustering algorithm has a concern of determining the best match.

B. Dash et al. [16] presented a hybrid block based motion estimation algorithm on the basis of JAYA algorithm. The main aim of this research work was to identify the global motion vectors in the reference video frame. In JAYA algorithm, the fitness estimation reduces the system complexity by estimating the fitness value of a search location. In this research study, the efficacy of the developed algorithm was tested on some real time videos like Akiyo, Stefan, container, kristen-sara, foreman, Johnny, old-town-cross, Carphone, football, and crowd-run. Experimental simulation shows that the developed hybrid algorithm attains decent performance in motion estimation in light of PSNR, computational time, SSIM and search efficiency. Still, the developed algorithm needs improvement in shift between dissimilar patterns that could be further exploited for the problem of under consideration.

H. Amirpour et al. [17] utilized temporally and spatially correlated neighbouring blocks for defining a dynamic

search pattern. In this literature, each video frame was divided into two blocks such as white and black blocks (like chess-board pattern). Initially, the motion vectors were obtained for the black blocks and then the white blocks were searched. Then, the motion vectors of white neighbourhood blocks were added to the current black blocks for better motion estimation. The experimental consequence shows that the developed approach attained better PSNR value related to other existing approaches. In contrast, the developed approach includes two concerns in motion estimation such as over-searching and trapped into a local minimum.

E. Cuevas [18] developed a new block matching algorithm based on Harmony Search (HS) optimizer for motion estimation. The developed algorithm utilizes the motion vectors as the potential solutions, which belongs to the search window. Then, the fitness function of HS optimizer estimates the matching quality of every motion vector. In this literature, the performance of the proposed algorithm was related with existing block matching algorithms in light of SSIM, and PSNR. Experimental consequence shows that the developed algorithm effectively handles the balance between computational complexity and coding efficiency. For the particle use, the developed algorithm achieved only considerable performance in motion estimation.

In order to highlight the above stated issues, a new cross square search pattern is proposed for improving the performance of motion estimation in videos.

3 Proposed algorithm

In computer vision application, motion estimation is an emerging research topic, which computes motion vectors by determining the pixel displacements in the video sequences. The motion estimation approaches are subdivided into two basic techniques such as block based

and pixel based motion estimation. In this research study, VSBM is proposed with cross square search pattern for motion estimation. The brief explanation about the undertaken approaches are detailed below.

3.1 Dataset description

In this research, the experiment is performed with a few standard video sequences such as Akiyo, Stefan, container, kristen-sara, foreman, Johnny, old-town-cross, Carphone, football, and crowd-run. *Dataset link* <https://media.xiph.org/video/derf/>. The explanation about every video sequence is detailed in Table 1. The sample frame of collected video sequences are stated in Fig. 1.

3.2 Variable size block matching

Generally, Fixed Size Block Matching (FSBM) is used for determining the motion of a block in the current video frame related to the reference video frame. But in FSBM, the size of the block is fixed (for instance, 16×16), and also the blocks are laid in a normal grid spanning that leads to blocking artifacts. It results in a high motion compensation error and also causes sharp spatial edges between the neighbourhood blocks in the motion compensation frame. In order avoid these concerns, VSBM is undertaken in this research study. In VSBM, the size of the block is variable (for instance, 64×64 , 32×32 , and 16×16) that decreases the overall complexity by finding whether the fitness value of a search location is to be estimated or not.

After collecting the video sequences, each frame is resized into 256×256 . In video coding, motion estimation and compensation are used for exploring the inter frame temporal correlation in the videos. In VSBM technique, the video frames are subdivided into non-overlapping block pixels $M \times M$, which is named as macro-blocks. For each macro blocks in the current frame, a best match block is identified with in a search region of the

Table 1 Details about collected video sequences

Video sequence	Motion characteristics	Frames	Frame size	Video format
Akiyo	Small motion and static background	300	176×144	QCIF
Stefan	High	300	352×288	CIF
Container	Gentle and smooth	300	176×144	QCIF
Kristen-Sara	Smooth	600	1280×720	HD
Foreman	Moderate	398	176×144	QCIF
Johnny	Smooth	600	1280×720	HD
Old-town-cross	High	500	1920×1080	HD
Carphone	Moderate	381	176×144	QCIF
Football	High	300	352×240	SIF
Crowd-run	High	500	1920×1080	HD

Fig. 1 Sample video frames, **a** foreman, **b** container, **c** Akiyo, **d** football, **e** Johnny and **f** Crowd-Run

size $x + 2W - 1, y + 2W - 1$, where W is indicated as maximum displacement. Hence, the matching procedure of the macro block is carried out based on the cost function. The cost function utilized in this research work is SSD that is mathematically indicated in Eq. (1).

$$J_{SSD}(x, y) = \sum_{(x,y)} W_R(x + 2W - 1, y + 2W - 1) - W_{R+1}(x + 2W - 1, y + 2W - 1)^2 \quad (1)$$

where, W is indicated as maximum displacement, (x, y) is denoted as image pixel location, R is stated as reference video frame, and $R + 1$ is represented as current video frame. If SSD of the current macro block is higher than the redefined threshold (except zero), the respective macro block is spilt into four small blocks. This procedure is repeated until the minimum error is obtained and then the cross square search pattern is applied in a micro block (with motion) to identify the motion vector. At last, the remerging procedure is performed for remerging the macro blocks that does not reduces the SSD and also improves the quality of image. Graphical illustration of block matching for 16×16 pixels and search parameter P of 7×7 is given in Fig. 2.

3.3 Cross square search pattern

The cross square search pattern utilizes temporal and spatial coherence in the surrounding blocks. In most of

Fig. 2 Graphical illustration of block matching for 16×16 pixels and search parameter size P of 7×7

the videos, motion in the current block is similar to the motion in the adjacent neighbourhood blocks. Then, Mean of Absolute Difference (MAD) is calculated for adjacent

neighbourhood blocks in the current frame by using Eq. (2).

$$J_{MAD(x,y)} = \frac{1}{N^2} \sum_{x=0}^{n-1} \sum_{y=0}^{n-1} |C_{xy} - R_{xy}| \quad (2)$$

where, N is represented as macro block size, R is indicated as reference video frame, and C is stated compensation video frame. This procedure is repeated until obtaining the minimum cost function. The location of minimum cost function is named as motion vector. In cross square search pattern, motion vector of the current block is same as the motion vector in the collocated block of the previous video frame (due to temporal connection) or the current block is similar to the neighborhood blocks in the current video frame (due to spatial connection). The graphical illustration of cross square search pattern is indicated in Fig. 3.

3.4 Step by step procedure of proposed algorithm

Step 1: Read the input video and convert the video into frames.

Step 2: Split the reference and current frames into blocks by using Eq. (3), where W is indicated as maximum displacement.

$$x + 2W - 1, y + 2W - 1, W = 32, 16, \text{ and } 8 \quad (3)$$

Step 3: Find the motion present in the macro blocks or not by using SSD that is mathematically denoted in Eq. (1).

Step 4: If block has motion then shrink block size to $W = 16$ and go back to step 2. Or else, go to step 8.

Step 5: Again, if block has motion then shrink block size to $W = 8$ and go back to step 2. Or else go to step 8.

Fig. 3 Graphical illustration of cross square search pattern

Step 6: Then, select the center pixel from step 4 and process 7×7 block using Eq. (4).

$$(x + 2W - 1 - 3, y + 2W - 1 - 3), \text{ where } W = 8 \quad (4)$$

Step 7: Apply search pattern in step 5 by utilizing the Eq. (5).

$$Pattern(x,y) = \begin{cases} Pixelvalue & dx = 0, 1, 2 \text{ and } dy = 0, 1, 2 \\ Pixelvalue & dx - 3, dy - 3 \\ Pixelvalue & dx + 3, dy + 3 \\ Pixelvalue & dx + 3, dy - 3 \\ Pixelvalue & dx - 3, dy + 3 \\ 0 & otherwise \end{cases} \quad (5)$$

Step 8: Find the MAD between reference and current block by using Eq. (2).

Step 9: Update the center pixel in step 6 and repeat the step 7 until obtaining minimum MAD that helps in finding the motion vector.

Step 10: Repeat the steps until all the motion vectors are processed.

4 Result and discussion

In this research, the exhaustive simulations are carried out by utilizing MATLAB 2018a with core i9 3.0 GHz, windows 10 operating system, and 16 GB RAM in order to validate the efficacy of the proposed algorithm. *Benchmark algorithms* To compare the efficiency of the proposed algorithm, many search algorithms are considered as the benchmark; dynamic pattern (square) [16], dynamic pattern (diamond) [16], dynamic pattern (hexagon) [16], and HS with block matching [18]. *Datasets undertaken* In this work, the experiments are performed with some of the standard and extensively utilized video sequences such as Akiyo, Stefan, container, kristen-sara, foreman, Johnny, old-town-cross, Carphone, football, and crowd-run. Hence, the details about undertaken video sequences are stated in Table 1. The sample motion estimated and compensation video frames are graphically denoted in Fig. 4.

4.1 Performance measure

Several performance measures such as PSNR, SSIM, and search count are utilized for representing a detailed comparative analysis about the performance of the proposed algorithm along with the benchmark or comparative approaches; dynamic pattern (square) [16], dynamic pattern (diamond) [16], dynamic pattern (hexagon) [16], and HS with block matching [18]. PSNR is used as a quality measurement between the reference frame $R(x, y)$ and

Fig. 4 **a** Current frame, **b** Reference frame, and **c** Compensation frame

compensation frame $C(x, y)$. Generally, PSNR is estimated using Mean Square Error (MSE) that is denoted in Eqs. (6) and (7).

$$MSE = \frac{1}{mn} \sum_{x=0}^{m-1} \sum_{y=0}^{n-1} \|R(x, y) - C(x, y)\|^2 \quad (6)$$

where, m and n are stated as image (single video frame) dimension, $R(x, y)$ is indicated as reference frame, and $C(x, y)$ is represented as compensation frame. The criterion utilized to evaluate the PSNR is given in Eq. (7).

$$PSNR = 20 \log_{10} \left(\frac{\max(R(x, y))}{\sqrt{MSE}} \right) \quad (7)$$

Additionally, the efficacy of the proposed algorithm is further analyzed by utilizing the performance metrics such as SSIM, and search count. SSIM is mathematically indicated in Eqs. (8).

$$SSIM(x, y) = \frac{(2\mu_x\mu_y + c_1)(2\sigma_{xy} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)} \quad (8)$$

where, σ and μ are represented as standard deviation and mean, and c_1 and c_2 are indicated as constants.

4.2 Quantitative investigation

This section detailed about the experimental results obtained with respect to the above-mentioned performance measures.

4.2.1 Quantitative investigation on the basis of PSNR value (dB)

In this segment, the performance of the proposed and benchmark algorithms are analysed in light of PSNR. By inspecting Table 2, it is observed that the proposed

Table 2 Performance analysis of proposed and existing algorithms in light of PSNR value (dB)

PSNR value (dB)						
Methods	Carphone	Akiyo	Container	Foreman	Stefan	Football
Dynamic pattern (square) [16]	31.51	35.51	43.17	32.24	25.9	23.06
Dynamic pattern (diamond) [16]	31.5	35.5	43.17	32.15	25.35	23.05
Dynamic pattern (hexagon) [16]	31.49	35.48	43.17	31.85	25.15	23.03
HS with block matching [18]	31.49	35.44	43.16	32.43	25.89	23.01
Cross square search pattern	39.57	50.40	43.82	37.64	28.058	28.921

algorithm attained better PSNR value in all the video sequences (foreman, Stefan, container, Carphone, football, and Akiyo) compared to the benchmark algorithms. In Carphone video sequence, the proposed cross square search pattern almost showed 8.05 dB improved in PSNR value compared to the benchmark algorithms such as dynamic pattern (square, diamond, and hexagon) [16] and HS with block matching [18]. From Table 2, it is observed that the proposed algorithm results in a maximum of 0.66 dB and 14.9 dB is improved as compared to the benchmark algorithms for the container and Akiyo video sequences. Additionally, the PSNR value of the proposed algorithm is superior compared to other benchmark algorithms in the video sequences; Foreman, Stefan, and Football. Graphical illustration of the proposed and benchmark algorithms in light of PSNR value is indicated in Fig. 5.

4.2.2 Quantitative investigation on the basis of search counts

This section details the number of search counts required for identifying the best matched macro blocks in the reference frame R . Therefore, the computational cost of

Fig. 5 Graphical illustration of proposed and benchmark algorithms in light of PSNR value (dB)

Table 3 Performance analysis of proposed and existing algorithms in light of average search count

Average number of search counts (%)						
Methods	Carphone	Akiyo	Container	Foreman	Stefan	Football
Dynamic pattern (square) [16]	6.85	2.14	2.38	8.44	14.26	21.47
Dynamic pattern (diamond) [16]	8.16	1.77	1.9	11.39	14.56	22.99
Dynamic pattern (hexagon) [16]	7.62	1.72	1.84	10.51	14.42	17.61
HS with block matching [18]	12.2	11.5	8	11.2	17.1	15.2
Cross square search pattern	3.71	2.85	3.73	4.007	4.085	3.99

proposed and the existing benchmark algorithms are analysed by means of the total number of searchers required to identify the most precise motion vectors. In Table 3, the performance of the existing benchmark and proposed algorithm are analysed in light of search counts. From Table 3, it is observed that the proposed algorithm requires only limited number of searchers to identify the accurate motion vectors as related to the benchmark algorithms like dynamic pattern (square, diamond, and hexagon) [16] and HS with block matching [18].

For example, in Carphone video, the search count of the proposed algorithm is 3.71%, which is minimum related to the benchmark algorithms. From Table 3, it is witnessed that the proposed cross square search pattern results in a maximum of 7.383%, 10.475%, and 19% degradation in search counts compared to other benchmark algorithms for the foreman, Stefan, and football video sequences. But in the case of container and Akiyo videos (slow motion sequences), the proposed algorithm achieved only reasonable performance due to the correlation of search pattern and magnitude of motion vector of a block. Meanwhile in slow motion video sequences, the local minima are trapped or it may carry out unnecessary intermediate searches. Hence, the graphical illustration of the proposed and benchmark algorithms in light of the average search count is specified in Fig. 6

4.2.3 Quantitative investigation on the basis of SSIM value

In this phase, the performance of the proposed and benchmark algorithms are investigated by means of SSIM value. It is utilized to verify the similarity between the reference frame R and compensation frame C . In Table 4, the SSIM of existing benchmark and proposed algorithm are investigated for the video sequences; Carphone, Akiyo, and foreman. By inspecting Table 4, it is observed that the proposed algorithm attained better SSIM value related to the existing benchmark algorithms; dynamic pattern search [16] and HS with block matching [18]. In most of the existing studies, fixed block size is used for motion estimation. In this research, variable block size is used along with cross square search pattern in order to improve the performance of motion estimation. The proposed algorithm decreases

Fig. 6 Graphical illustration of proposed and benchmark algorithms in light of average search count

Table 4 Performance analysis of proposed and existing algorithms in light of SSIM value

SSIM value			
Methods	Carphone	Akiyo	Foreman
Dynamic pattern search [16]	0.9761	0.9863	0.9849
HS with block matching [18]	0.9764	0.9866	0.9847
Cross square search pattern	0.9841	0.9996	0.9943

the overall complexity by identifying whether the fitness value (SSD) of a specific search location is to be estimated or not. Graphical illustration of the proposed and benchmark algorithms in light of SSIM is specified in Fig. 7.

5 Conclusion

In this research paper, VSBM with cross square search pattern is proposed for motion estimation in the video sequences. For every macro block, the proposed algorithm significantly identifies the motion vectors, which results in minimum prediction error. The major benefit of proposed algorithm is that it enables macro blocks to be divided along with motion boundary. Compared to other benchmark algorithms, the proposed algorithm achieved good performance in motion estimation in light of SSIM, PSNR, and average search count. From the experimental study, the proposed algorithm results in a maximum of 14.9 dB improved in PSNR as compared to the benchmark algorithms. Meanwhile, the proposed algorithm results in a maximum of 19% degradation in search counts related to other algorithms. In this research, the complexity of the

Fig. 7 Graphical illustration of proposed and benchmark algorithms in light of SSIM

proposed algorithm is estimated using $O(N^4 \log_2 N)$, where N is indicated as number of micro blocks. In future work, the proposed algorithm is applied in hardware based implementation for real time applications.

Compliance with ethical standards

Conflict of interest On behalf of all authors, the corresponding author states that there is no conflict of interest.

References

- Mukherjee R, Vinod IG, Chakrabarti I, Dutta PK, Ray AK (2020) Hexagon based compressed diamond algorithm for motion estimation and its dedicated VLSI system for HD videos. *Expert Syst Appl* 141:112919
- Yang D, Zhong X, Gu D, Peng X, Hu H (2020) Unsupervised framework for depth estimation and camera motion prediction from video. *Neurocomputing* 385:169–185
- Shukla D, Jha RK (2016) Robust motion estimation for night-shooting videos using dual-accumulated constraint warping. *J Vis Commun Image Represent* 38:217–229
- Kim SW, Yin S, Yun K, Choi JY (2014) Spatio-temporal weighting in local patches for direct estimation of camera motion in video stabilization. *Comput Vis Image Underst* 118:71–83
- Bare B, Yan B, Ma C, Li K (2019) Real-time video super-resolution via motion convolution kernel estimation. *Neurocomputing* 367:236–245
- Bhattacharjee K, Kumar S, Pandey HM, Pant M, Windridge D, Chaudhary A (2018) An improved block matching algorithm for motion estimation in video sequences and application in robotics. *Comput Electr Eng* 68:92–106
- Cabon S, Porée F, Simon A, Ugolin M, Rosec O, Carraut G, Pladys P (2017) Motion estimation and characterization in premature newborns using long duration video recordings. *IRBM* 38(4):207–213

8. Puthenpurayil SP, Chakrabarti I, Viridi R, Kaushik H (2016) Very large scale integration architecture for block-matching motion estimation using adaptive rood pattern search algorithm. *IET Circuits Devices Syst* 10(4):309–316
9. Chen BH, Kopylov A, Huang SC, Seredin O, Karpov R, Kuo SY, Lai KR, Tan TH, Gochoo M, Bayanduuren D, Gong CS (2016) Improved global motion estimation via motion vector clustering for video stabilization. *Eng Appl Artif Intell* 54:39–48
10. Charles PK, Habibulla Khan DK (2017) A novel search technique of motion estimation for video compression. *Glob J Comput Sci Technol* 17(2):1–7
11. Hemanth DJ, Anitha J (2018) A pattern-based artificial bee colony algorithm for motion estimation in video compression techniques. *Circuits Syst Signal Process* 37(4):1609–1624
12. Bao W, Lai WS, Zhang X, Gao Z, Yang MH (2019) MEMC-Net: motion estimation and motion compensation driven neural network for video interpolation and enhancement. *IEEE Trans Pattern Anal Mach Intell*. <https://doi.org/10.1109/TPAMI.2019.2941941>
13. Betka A, Terki N, Toumi A, Hamiane M, Ourchani A (2019) A new block matching algorithm based on stochastic fractal search. *Appl Intell* 49(3):1146–1160
14. Lin L, Wey IC, Ding JH (2016) Fast predictive motion estimation algorithm with adaptive search mode based on motion type classification. *SIVIP* 10(1):171–180
15. Wu M, Li X, Liu C, Liu M, Zhao N, Wang J, Wan X, Rao Z, Zhu L (2019) Robust global motion estimation for video security based on improved k-means clustering. *J Ambient Intell Human Comput* 10(2):439–448
16. Dash B, Rup S, Mohanty F, Swamy MNS (2019) A hybrid block-based motion estimation algorithm using JAYA for video coding techniques. *Digital Signal Process* 88:160–171
17. Amirpour H, Ghanbari M, Pinheiro A, Pereira M (2019) Motion estimation with chessboard pattern prediction strategy. *Multimed Tools App* 78(15):21785–21804
18. Cuevas E (2013) Block-matching algorithm based on harmony search optimization for motion estimation. *Appl intell* 39(1):165–183

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.