Proceedings Paper DOI: 10.58190/icisna.2024.82

PROCEEDINGS OF

INTERNATIONAL CONFERENCE ON INTELLIGENT SYSTEMS AND NEW APPLICATIONS

https://proceedings.icisna.org/

2nd International Conference on Intelligent Systems and New Applications (ICISNA'24), Liverpool, April 26-28, 2024.

Object Tracking using Modified Hexagonal Search Algorithm

Khalid Ghoul[1], Sofiane Zaidi[1], Zakaria Laboudi[2]

[1] *Department of Mathematics and Computer Science, Research Laboratory on Computer Science's Complex Systems (RELA(CS)2), University of Oum El Bouaghi, Algeria*

khaled.ghoul@univ-oeb.dz, ORCID: 0000-0002-8045-2810

Zaidi.sofiane@univ-oeb.dz, ORCID: 0000-0002-5743-8850

[2]*Department of Networks and Telecommunications, University of Oum El Bouaghi, Algeria*

laboudi.zakaria@univ-oeb.dz, ORCID: 0000-0002-3699-4081

***Abstract*— Object tracking is one of the most important problems in computer vision, so many methods have been proposed to solve it. The principle of object tracking mainly relies on motion estimation to track the motion of objects. In motion estimation, a variety of algorithms based on block matching have been proposed in order to address diverse issues such as reducing the number of search points, computational complexity. Although existing methods for object tracking provide good results, most of them face performance issues related to computation time. In this work, we propose a modified hexagonal search algorithm (MHS) to improve computation time of estimated motion while preserving efficiency. The proposed algorithm proceeds in three steps. In the first step, a small hexagonal pattern is used to find the smaller motion vectors. In the second step, the large hexagonal pattern is used to determine the direction of motion. In the third step, the small hexagonal search pattern is used to find the final solution. The MHS algorithm is used within the object tracking system after performing the object detection step. To validate our proposal, we consider several video sequences. The experimental results show that MHS outperforms some related works.**

***Keywords*— Object tracking, Motion estimation, Block Matching, Hexagonal search algorithm, Object detection**

I. Introduction

In recent years, Object Tracking (OT) has attracted much interest due to their various application areas, especially in video surveillance [1]-[2], action and activity recognition [3], augmented reality [4] and robotics [5]. However, despite the significant improvements made to tracking algorithms, the OT problem still remains open due to the fact that these algorithms relate to many environmental factors.

OT involves estimating the trajectory of an object in the 2D plane of an image in the same way as if it moves in the real 3D scene. This task requires locating each object from one image to the next one. Many tracking techniques predict the position of an object in a frame based on the motion of the object observed in previous images. Each detected object must be associated with the corresponding object in the next frame to update its trajectory; otherwise a new trajectory is created [6]. OT is often difficult due to the complex shapes of the objects, their non-rigid nature, motion, partial or complete occlusion, changes in scene lighting, etc. These can be simplified by simple assumptions, such as smooth motion and prior knowledge of the number, size, shape, and appearance of objects. OT allows extracting additional features: trajectory, speed, direction of motion, and position at a precise moment [6]. The process of tracking moving objects generally comprises three stages: detection of moving objects, classification of these objects, and finally tracking [7].

Motion estimation algorithms can be used in the tracking process either in the detection stage or in the tracking stage, in particular Block Matching Algorithms (BMA). These algorithms locate similar blocks in an image sequence, i.e., search for the best match between two blocks located in two consecutive frames, in order to estimate the motion or displacement vector, the aim of which is to eliminate temporal redundancy. The main disadvantage of BMA is their computational cost, which is caused by the large search space. To deal with this drawback, many algorithms have been proposed; as examples we cite: New Three-Step Search [8], Diamond Search [9], Star Diamond Search [10], Hexagonal Search [11], and Cross-Hexagonal Search (CHS) [12].

In this paper, we propose a novel approach for OT by using block-matching motion estimation algorithms. The developed system uses the background subtraction algorithm in the detection step and our novel Hexagonal Search Algorithm the in the tracking step. The primary goal is to reduce the number of points searched to make the tracking process faster. Thus, our proposal mainly consists in a Modified Hexagonal Search algorithm (MHS) that attempts to improve the BM algorithm's performance, in particular the Hexagonal Search algorithm, in terms of computation complexity. The proposed algorithm proceeds in three steps. In the first step, a small hexagonal pattern is used to find the smallest motion vector, thus finding fewer search points. The second step consists to use a large hexagonal pattern to identify the direction of the motion vector. In the third step, we use the small hexagonal search pattern to locate the final solution. The computation complexity is decreased by using a smaller hexagonal search pattern in the first step for the MHS, which allows for a quicker search for the minimal motion vector.

The paper is organized as follows. In Section 2, we review some relevant related works in the literature. In Sections 3 and 4 we present the Hexagonal and cross hexagonal algorithms. In Section 5, we give details of our proposed system for OT. Tests and results are summarized and discussed in section 6. Finally, in section 7, we draw some conclusions and perspectives

II. RELATED WORKS

Many BM algorithms have been developed [13], the simplest being the exploration of all blocks of the search window called Full Search [14].

C. Zhu et al. proposed Hexagonal Search (HS) in 2002 [11]. This algorithm will be described in the next section.

The Cross Hexagonal Search (CHS) algorithm was proposed by S. Zhu et al [12]. In this algorithm, two Cross-Shaped Search Patterns are applied in the first and second step. Then, a Large Cross-Search Pattern is applied. Next, a Large Hexagonal Search Pattern is applied repeatedly until the minimum block distortion (MBD) point is at the center. Finally, the SHSP is applied in order to find the final motion vector.

The Kite Cross hexagonal search was proposed by Hamood and Abdulmunem [15]. This algorithm uses different search patterns such as kite, cross, and hexagonal in order to find the best motion vector.

R. Mukherjee et al. [16] introduced an algorithm called Hexagon Based Compressed Diamond Algorithm that attempts to reduce the complexity by lowering the number of search points in the process. The algorithm uses an early termination technique and an adaptive search pattern that can uniformly deal with slow and fast motion content in a video.

P. Av et al [17] proposed an all-direction search pattern, which searches for the best block in all possible directions.

III. HEXAGONAL SEARCH ALGORITHM

The algorithm uses two search patterns as illustrated in Fig. 1: a Large Hexagonal Search Pattern (LHSP) and a Small Hexagonal Search Pattern (SHSP).

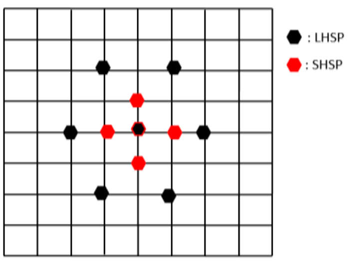

Fig. 1 Search pattern model for the HS algorithm.

The HS algorithm is summarized as follows [11]:

Step 1: LHSP is applied at the center of the search window; if the MBD point is the center point, then go to Step 3; otherwise, go to Step 2.

Step 2. LHSP is applied at the MBD point obtained in the Step 1. If the new MBD point is located at the center position, proceed to Step3, otherwise, keep on repeating this step.

Step 3. Switch the LHSP to SHSP. The MBD point found in this step is be the best matching block.

IV. MODIFIED HEXAGONAL ALGORITHM

Our proposed algorithm consists in modifying HS in such a way that the SHSP is applied in the first step. This is because in real-world video sequences, more than 80% of the blocks can be regarded as stationary [10], [18]-[20]. Therefore, the MHS is defined as follows:

Step1: SHSP with 5 points of search is applied at the center of the search window. If the MBD point is the center point, then the search stops; otherwise, go to Step 2;

Step2: LHLP with 7 points of search is applied at the MBD point obtained in the previous step. If the new MBD point is the center point, then go to Step 3; otherwise repeat this step;

Step3: SHSP is applied and the MBD point is found. This point is considered as the motion estimation solution (final solution).

The block diagram of the proposed system is shown in Fig.2.

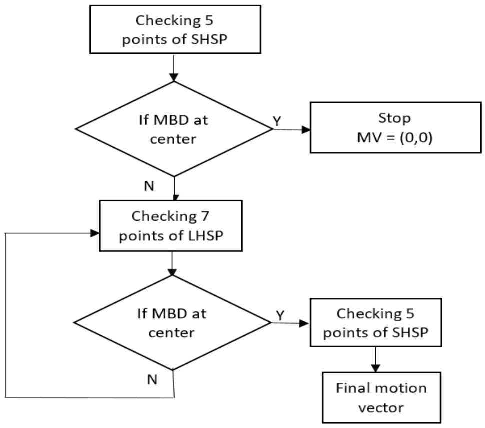

Fig. 1 Block schematic of the Modified HS (MHS)

V. OBJECTS TRACKING SYSTEM

The objects tracking system consist of the following phases:

A. Pre-processing Phase

Before performing any video processing operation, the quality of the image is crucial. At this stage, consideration should be given to improving the quality of the frames.

B. Object Detection Phase

On all frames of the video sequence, the background subtraction method for object detection and classification is applied as follow:

- Take an image background as reference frame denoted by I(t) and take the next frames as current frame denoted by I(t+1) to compare it with I(t).
- By relying on some simple arithmetic calculations, we segment out the objects through image subtraction technique, as follow :

$$f(t) = I(t+1) - I(t) \quad (1)$$

- A threshold " φ " is put on this difference image to improve the subtraction

$$f(t) - f(t+1) > \varphi \quad (2)$$

C. *Motion Estimation Phase*

In this step the MHS algorithm is applied as described above.

D. *Tracking Object Phase*

As a result of the previous phase, a motion vector $\vec{v} = (v_x, v_y)$is created; it allows keeping track the motion of a block in different positions. Using all motion vectors the object is tracked in the frames of the video. The block diagram of the proposed system is shown in Fig. 3.

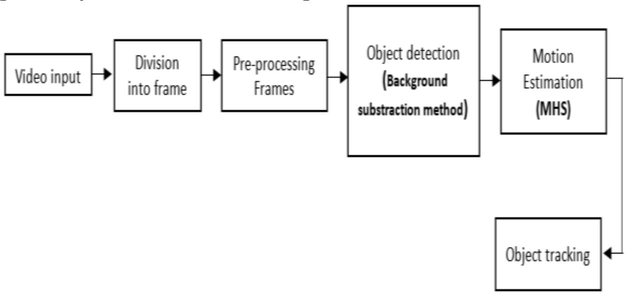

Fig. 3 Block Diagram of Object Tracking System

VI. Tests and Results

In this section, we evaluate the MHS algorithm according to the following criteria: computation complexity, prediction quality and computing time.

A. *Test sequences*

The experiments were carried out on color videos (i.e. Biker, vision Traffic, Trium, Ball, Jeux, Man and Dancer2 sequences). This dataset can be obtained from online resources (http://cvlab.hanyang.ac.kr/tracker_benchmark/datasets.html). All video sequences used for testing contain a single object moving in all frames. Table I presents the video sequences and the characteristics of each sequence.

TABLE I
VIDEOS SEQUENCES WITH SPECIFICATION USED IN EXPERIMENTS

N°	**Name**	**Number Frames**	**Size**
1	Ball	23	360×480
2	Baker	142	360×640
3	Atrium	61	360×640
4	Dancer2	150	262×320
5	Jeux	116	640×630
6	Man	50	193×241
7	Vision traffic	72	360×640

B. *Performance Evaluation Criterion*

Now, we compare our MSH algorithm with the two block matching methods HS, and CHS. Datasets consist of a single object moving in all frames. Moreover, for all video datasets used, the camera is stationary. The HS and CHS algorithms are executed on a 16x16 macro block, with a window size of -7 to +7. Motion vectors are found using Mean Absolute Difference (MAD) given in Equation (3) as a block matching criterion.

$$MAD(x, y) = \frac{1}{N^2}\sum_{l=0}^{N-1}\sum_{l=0}^{N-1}|I(x+k, y+l) - g(k,l)| \quad (3)$$

where the values $I(k,l)$ and $g(k,l)$ denote the luminance values of I and g (image), and (x, y) is the candidate vector.

1) *Computation complexity*

The comparison between the proposed algorithm and the other algorithms is carried out by calculating the mean of the average number of search points NSP_{avg} required to find the motion vector for each frame. The obtained results are summarized in Table II.

TABLE III
PERFORMANCE COMPARISON IN TERMS OF NSP_{AVG}

Algorithms Sequences	**MHS**	**HS**	**CHS**
Ball	5,02	10,66	8,90
Biker	5,31	10,79	9,20
Atrium	4,63	10,79	9,09
Dancer2	5,69	10,71	9,46
Jeux	5,10	10,83	9,02
Man	5,99	10,57	9,64
Vision traffic	5,25	10,79	9,15

Fig. 4 plots the average number of searching points per frame using 'Ball sequence', in each frame.

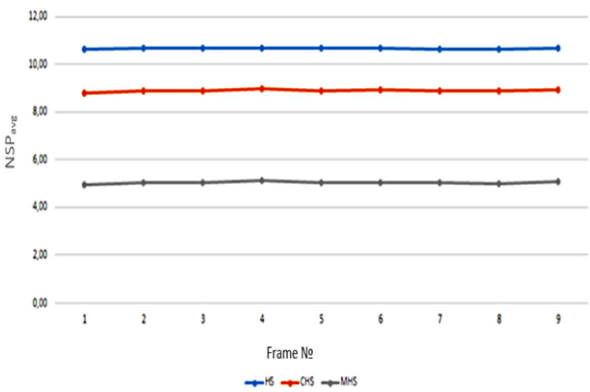

Fig. 4 Performance comparison of different algorithms in terms of NSP_{avg} for Ball sequence

2) *Prediction Quality*

The criteria $PSNR_{avg}$ is used to measure the average quality of all reconstructed images [21]. The corresponding Peak Signal to Noise Ratio (PSNR) value is calculated according to Equation (4):

$$PSNR = 10log_{10}\left(\frac{f^2}{MSE}\right) \quad (4)$$

where f is the maximum pixel value and MSE is the mean square error. The Mean Square Error (MSE) is described as shown in Equation 5.

$$MSE = \frac{\sum_{i=1}^{n}\sum_{j=1}^{m}(g_t(i,j)-\hat{g}_t(i,j))^2}{n\times m} \quad (5)$$

where $g_t(i,j), \hat{g}_t(i,j)$ and $n \times m$ represent the current frame, the reconstructed frame, and the frame size, respectively.

The results obtained are given on Table III.

TABLE IIIII
PERFORMANCE COMPARISON IN TERMS OF $PSNR_{avg}$

Algorithms Sequences	**MHS**	**HS**	**CHS**
Ball	25,53	25,46	25,49
Biker	29,85	29,77	29,77
Atrium	25,09	25,00	25,04
Dancer2	26,63	26,22	26,47
Jeux	31,42	31,06	31,18
Man	21,81	21,71	21,80
Vision traffic	31,91	31,07	31,73

3) *Computing time*

The calculation time is measured in seconds for each algorithm. Regarding the implementation, we used the Matlab programming language version R2017a under hardware architecture endowed with 2 cores, each core is 2.16 GHz (Intel(R) Celeron(R) CPU N2840). The results obtained are given on Table IV.

TABLE IVV
PERFORMANCE COMPARISON IN TERMS OF COMPUTING TIME (SECOND)

Algorithms Sequences	MHS	HS	CHS
Ball	0,11	0,21	0,15
Biker	0,14	0,23	0,20
Atrium	0,28	1,19	0,70
Dancer2	0,09	0,13	0,13
Jeux	0,32	0,48	0,42
Man	0,03	0,05	0,04
Vision traffic	0,14	0,24	0,22

C. *Results of Object Tracking*

The implementation of the tracking system described in section V allows us to obtain the results shown in Fig.5. The final tracked object is indicated by the green bounding box. The first ten frames of the video “Vision Traffic” are treated; the obtained results are shown in Fig.6.

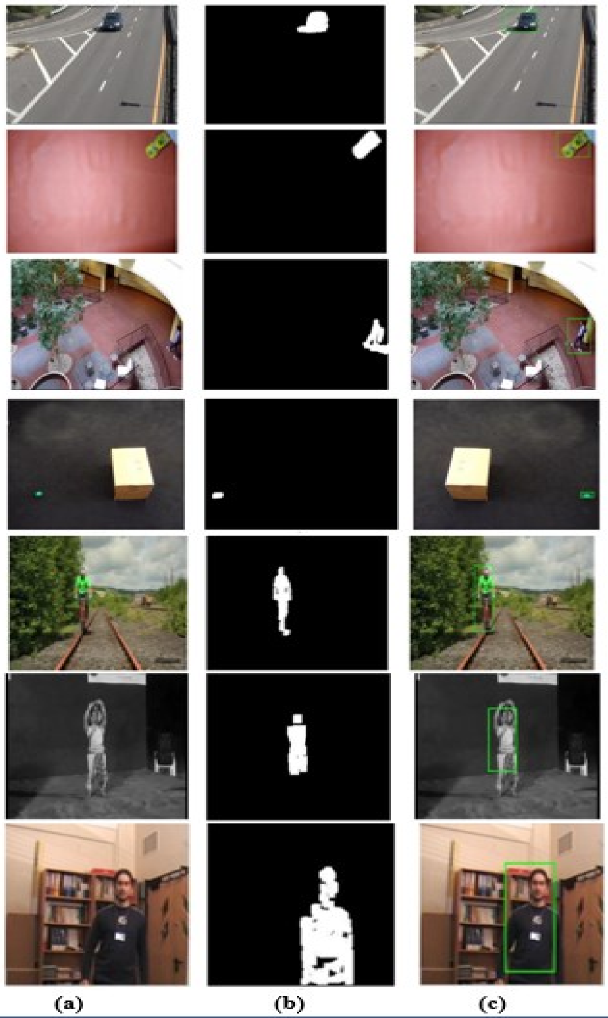

Fig.5 (a) Original video Frame (b) Background Subtraction (c) Object Tracking Result of the video sequences.

Fig.6 Result of object tracking using the 10 first frames of ‘Vision Traffic’ sequence

VII. Analysis and Discussion

According to Table II, one observes that the MHS is the best in terms of average number of search points for all video sequences compared to HS and CHS algorithms. This is because the MHS uses in the first stage the SHSP with 5 search points instead of 7 points for HS and 9 search points for CHS. Compared with HS and CHS, MHS saves approximately 3.88 to 5.64 research points. For instance, for the "Ball" video sequence, the MHS algorithm records 3.88 to 5.02 search points per frame, compared to CHS and HS, respectively.

Modified Hexagonal Search algorithm is qualitatively better in term of prediction quality as it has the highest PSNR compared to other methods (see Table III). This is due to the small search area (5x5) used in the first step of MHS. For example, for the 'Vision traffic' video sequence the $PSNR_{avg}$ is equal to: 30,91db for MHS, 31,73db for CHS and 31,07db for HS.

According to Table IV, the MHS outperforms the HS and CHS in terms of computation time measurements. Indeed, in the best case, the MHS requires 5 search points, while the HS and CHS require 9 and 11 search points, respectively.

The MHS is used to track an object by locating similar blocks in the video sequences. The goal of BMA is to find a matching block between the frames. Block matching involves partitioning the current frame into blocks and comparing each block with the corresponding block in the next frame. Then, a vector is created so that it maps the motion of a block from one position to another in a video sequence. Finally, these motion vectors provide displacements in the block, which can be used for object tracking. Figures 5 and 6 shows the tracking results.

VIII. Conclusions

In this work, we have proposed a modified hexagonal search algorithm and used it to track the motion of object in video sequences. The MHS has good adaptability to estimate the smallest motion vectors.

The experimental results showed that the MHS algorithm could make motion process estimation faster and reduce computation complexity. The comparative study proved that the MHS algorithm reduced the number of searched points per block compared to HS and CHS algorithms

As a future project, we plan to combine our algorithm with deep learning approaches to develop real-time object tracking systems.

References

[1] O. Elharrouss, N. Almaadeed, and S. Al-Maadeed, "A review of video surveillance systems". *Journal of Visual Communicat/ion and Image Representation*, vol. 77, pp. 103116, 2021.

[2] C. Kumar, S. Singh, "Security standards for real time video surveillance and moving object tracking challenges, limitations, and future: a case study". *Multimedia Tools and Applications*, pp. 1-32, 2023.

[3] Z. Hussain, Q. Z. Sheng, and W.E. Zhang, "A review and categorization of techniques on device-free human activity recognition". *Journal of Network and Computer Applications*, vol. 167, pp. 102738, 2020.

[4] Z. Makhataeva, A. H. Varol, "Augmented reality for robotics: A review", *Robotics*, vol. 9, no 2, pp. 21, 2020.

[5] J. Wilson, M. C. Lin, "Avot: Audio-visual object tracking of multiple objects for robotics". In: *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE. pp. 10045-10051, 2020.

[6] W. Luo, J. Xing, A. Milan, X. Zhang, W. Liu, and T.K. Kim, "Multiple object tracking: A literature review. Artificial intelligence", *Artificial intelligence*, vol. 293, pp. 103448, 2021.

[7] X. Zhou, V. Koltun, and P. Krähenbühl, "Tracking objects as points". In: European conference on computer vision. Cham: Springer International Publishing, pp. 474-490, 2020.

[8] S. Zhu, C. Zhang, "A fast algorithm of intra prediction modes pruning for HEVC based on decision trees and a new three-step search". *Multimedia Tools and Applications*, vol. 76, pp. 21707-21728, 2017.

[9] S. Zhu, K. K Ma, "A new diamond search algorithm for fast block-matching motion estimation". IEEE transactions on Image Processing, 2000, vol. 9, no 2, pp. 287-290.

[10] D. Kerfa, M. F. Belbachir, "Star diamond: an efficient algorithm for fast block matching motion estimation in H264/AVC video codec". Multimedia tools and applications, vol. 75, pp. 3161-3175, 2016.

[11] C. Zhu, X. Lin, and L.P. Chau, "Hexagon-based search pattern for fast block motion estimation". IEEE transactions on circuits and systems for video technology, vol. 12, no 5, pp. 349-355, 2002.

[12] S. Zhu, J. Tian, X. Shena, and K. Belloulata, "A Novel Cross-Hexagon Search Algorithm Based on Motion Vector Field Prediction", IEEE International Symposium on Industrial Electronics (ISIE), pp. 1870-1874, 2009.

[13] H.A Choudhury, M. Saikia, "Block matching algorithms for motion estimation: a performance-based study". Advances in Communication and Computing, pp. 149-160, 2015.

[14] L. C. Manikandan, S. A. H, Nair, K. P. Sanal Kumar, and R. K. Selvakumar,. A study and analysis on block matching algorithms for motion estimation in video coding. Cluster Computing, vol. 22, no Suppl 5, pp. 11773-11780, 2019.

[15] T. Z. Hamood, M. E. Abdulmunem, "A novel kite cross hexagonal search algorithm for fast block motion estimation," In: Journal of Physics: Conference Series. IOP Publishing, pp. 012026, 2021.

[16] R. Mukherjee, I. G. Vinod, I. Chakrabarti, P.K. Dutta, and A. K. Ray, "Hexagon based compressed diamond algorithm for motion estimation and its dedicated VLSI system for HD videos". Expert Systems with Applications, vol. 141, pp. 112919, 2020.

[17] P. AV, N. K. Darimireddy, and S. Siripurapu. "All Directional Search Motion Estimation Algorithm". Electronics, vol. 11, no 22, pp. 3736, 2022.

[18] K. Zhu, K. Ma, "A new diamond search algorithm for fast block-matching motion estimation," IEEE transactions on Image Processing, vol. 9, no 2, pp. 287-290, 2000.

[19] L. M. Po, W. C. Ma, "A novel four-step search algorithm for fast block motion estimation," EEE transactions on circuits and systems for video technology, vol. 6, no 3, pp. 313-317, 1996.

[20] D. Kerfa, A. Saidane, "An efficient algorithm for fast block matching motion estimation using an adaptive threshold scheme". *Multimedia Tools and Applications*, vol. 79, no 33, pp. 24173-24184, 2020.

[21] B. Zhang, Y. Zhang, B. Wang, X. He, F. Zhang, and X. Zhang, "Denoising swin transformer and perceptual peak signal-to-noise ratio for low-dose CT image denoising". *Measurement*, pp. 114303, 2024.