RESEARCH ARTICLE OPEN ACCESS

# An Efficient Adaptation of Edge Feature-Based Video Processing Algorithm for Wireless Multimedia Sensor Networks

Thanaa Jbeily [1], Mothanna Alkubeily [2], Iyad Hatem [3]
Thesis Student [1], Associate Professor [2], Assistant Professor [3]
Department of Mechanical and Electrical Engineering
Tishreen University
Lattakia - Syria

## ABSTRACT
Multimedia communication in wireless sensor networks is a very challenging task with the critical constraints of these networks. Motion estimation part is the heart of a video compression/coding standard, it can significantly improve video coding efficiency by reducing temporal redundancy existing in a video sequence. It depends on the total number of search points for finding the motion vector in the search window. In this paper, we firstly detail the block matching algorithms and present our own classification of fast block matching algorithms. Secondly, we introduce our proposal of adopting an adaptive motion estimation algorithm which is more efficient for WMSNs. Finally, we compare between the considered algorithm using in WMSN and our proposal to adopt another algorithm depending on different parameters as Peak Signal to Noise Ratio (PSNR), and compression ratio applied on MATLAB platform. In addition and based on the importance of power consumption over WMSNs, we propose a new method to compare between the different block matching algorithms based on it.

***Keywords:-***Wireless Multimedia Sensor Networks, Edge Feature-Based Video Coding, Motion Estimation, Adaptive search pattern, block-matching algorithm.

## I. INTRODUCTION

Wireless Multimedia Sensor Networks (WMSNs) [1,2] are a logical evolution of traditional scalar Wireless Sensor Networks (WSNs) in order to support new promising application scopes such as visual environment monitoring, object tracking, visual investigating and recognition, and other inapplicable domains over WSN. A WMSN consists of a group of tiny size, both self organized and powered sensory devices called multimedia sensor nodes. Those nodes are provided by special equipment like cameras and microphones to enable picking up the surrounding phenomenon related multimedia data without the requirement of direct contact with that phenomenon. Sensor nodes send the multimedia data (video, images, audio, and scalar data) to a base station called the sink, which, in its order, delivers the data to the user over the internet or satellite communications without direct human intervention in the monitoring area.

WMSNs has been in the limelight in recent years due to its two basic keys: (1) wireless communication, i.e. reaching the deserved information at anytime and anywhere without a pre-existent infrastructure, and (2) multimedia support, i.e. getting more extensive and accurate information to provide new applications or to develop the current ones. However, WMSNs face many challenges [3,4], that we can classify into two categories, old challenges that already were present in WSNs but they are settled due to the bulky nature of multimedia data comparing with scalar data, and the new challenges produced from the unique characters of multimedia data which require a special deal to be captured. Different proposals were introduced to overcome the past challenges, and we can say that each one of them is related to one or more of the following four basic classes: (1) Routing protocol, (2) Network architecture, (3) Data acquisition style, and (4) Data processing. WMSN related researches show the importance of data processing. Indeed, consumed resources caused by data processing not only approximate the consumed resources caused by data transmission, but also affects on it [5,6]. We can classify data processing techniques over WMSN to: (1) Feature extraction [7], (2) Data coding [8], and (3) Data Fusion [9]. Traditional video processing techniques are inapplicable over WMSNs. Therefore, the researchers aim to find special processing algorithms which offer simplicity with low power consumption on a hand and effective compression rates with acceptable data quality on the other hand. Edge feature-based video processing algorithm over WMSN proposed in [10] is an example of those algorithms. It depends on the processing technique of feature extraction containing edges detection and data coding containing both distributed and individual resource coding [5,6,8].

The reminder of this paper is organized as follows: In section II, we study the considered algorithm in WMSN: detail its energy consumption balancing style, light on its basic issues specially its inefficiency motion estimation algorithm, and also put forward our own broad classification of fast block matching algorithms. In section III, we introduce our proposal of adopting an adaptive motion estimation algorithm whose characters is more efficient for WMSNs. In section IV, we evaluate the impact of the two algorithms depending on different parameters, and we propose a new power consumption method to compare between the different block matching algorithms. Finally, conclusion and future work are presented in section V.

## II. RELATED WORKS

### *A. Edge Feature-Based Video Processing Algorithm Over WMSN (The considered Algorithm)*

Authors in reference [10] puts forward edge feature-based video processing algorithm over WMSN. It is an effective and low-complexity algorithm that exploits the changes of objects' edges among the video frames to determine the motion regions. They introduced a distributed coding in order to balance energy consumption by sharing the video processing tasks over three multimedia sensor nodes along a path from a source node to a cluster-head. We can divide the considered algorithm into the following five basic phases.

1. *Detecting the edges of objects:* There are several edge detection algorithms [11,12]. However, homogeneity edge detection technique was used in the considered algorithm. It is a suitable technique to implement on WMSN due to its characteristics such as low complexity, low error rate, and low execution time [11].
2. *finding the edges of objects changes among frames*: The captured video data consist of a sequence of frames. The considered algorithm deal with the picked frames as one of the two following frame types [10]: (1) Intra-frame (I-frame) which stores only information within the current frame, and (2) Predictive-frame (P-frame) which stores the difference among one or more neighboring frames. If the input is an I-frame, it is directly encoded as in the fifth phase. On the other hand, if it is P-frame, a comparing process between objects edges of the current frame and the previous reference frame will take a place. Comparing process aims to find the difference of objects edges between two frames[10].
3. *Active regions marking phase:* This phase depends on related results of the previous phase to mark the active regions .
4. *Motion estimation phase:* There are several motion estimation algorithms [13-15]. However, Three Step Search-TSS algorithm [14] was used in this algorithm to estimate and find the motion vectors.
5. *Encoding phase:* I-frames are encoded by conventional MPEG-2/H.262 encoder that is suitable for wireless application [15]. While, P-frames are encoded using Wyner-Ziv distributed coding [8,16] and JPEG standard [17].

### *B. Energy Consumption Balancing Style*

In order to balance energy consumption, the considered algorithm shares its basic tasks included edge detection, motion estimation, and coding (among three multimedia sensor nodes: source node S, transformation node T, and coding node C [10]). According to the frame type, we can recognize the two following cases.

1) I-frames: They are encoded directly along a path from a source node S to a cluster-head node CH as the message sequence chart depicts in Fig. 1.
2) P-frames: As the message sequence chart illustrates in Fig. 2, P-frames are encoded using Wyner-Ziv distributed coding to reduce the load on the source node S and share it with the node T.

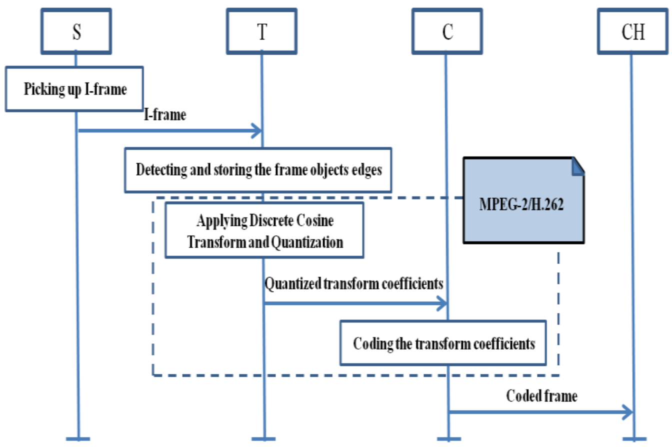


Fig. 1. Message sequence chart of I-frame processing .

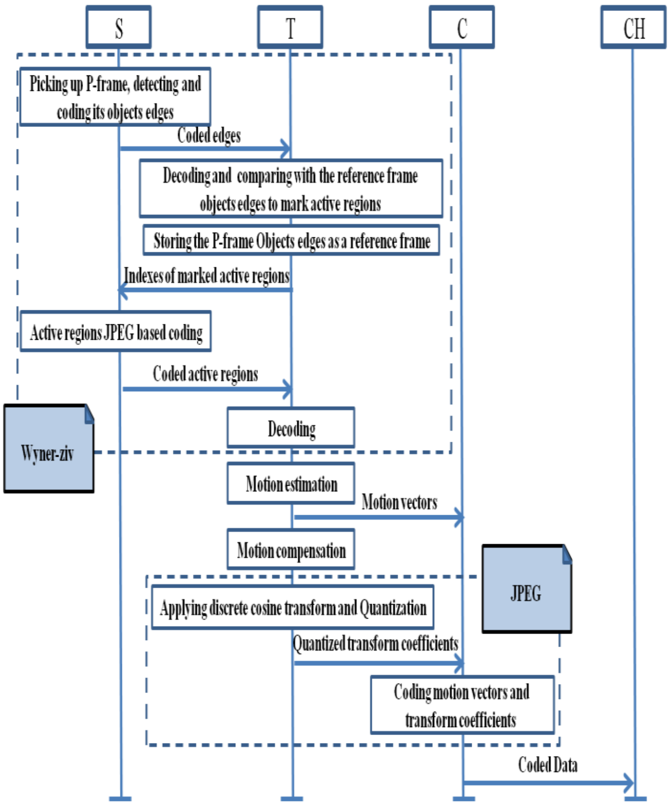


Fig. 2. Message sequence chart of P-frame processing

### C. Basic Issues of the Considered Algorithm

Despite of the distinct characteristics which are offered by this algorithm as energy consumption balancing and the quality improvement of decoding video by using objects edges. It still faces, from our views, several issues which we can summarize as following.

- It used only I-frames and P-frames without bidirectional frames in video compression, thus the rate of compression is not optimized.
- I-frames are used as reference frames to enhance the video quality and avoid the errors accumulation. Nevertheless, they are encoded depending on MPEG-2/H.262 encoder. As a result, the quality and compression rate of encoding frames have not exploited.
- Good results are offered only when the algorithm is applied on slow motion changes video sequences.
- The number of hops between the source node S and the cluster-head CH is three hops. Therefore, failing one of these nodes, i.e. T or C, leads to remove its tasks to the source node. Consequently, the source node will be exhausted.
- Despite of tasks sharing over S,T and C, this sharing is not evenhanded. Node T has to achieve both the Discrete Cosine Transform-DCT and the motion estimation tasks. All the related researches focused on the fact that the most computationally expensive and resource hungry operation in the entire compression process is the motion estimation [14,15,18]. Thus, this field has seen our highest attention.

### D. Motion Estimation

In spite of the motion estimation deep complexity, it is the heart of video processing [13,15,19]. It benefits of the high correlation among successive video frames and exploits their information redundancy - called temporal domain redundancy. It determines the shift of a particular region in the current frame considering with a suitable region in a reference frame. This shift is represented by displacement vector which is commonly known by motion vector-MV [13,14]. Depending on the particular region whose motion vector is needed to be found, motion estimation algorithms are classified into two basic classes [20]: (1) Pixel based algorithms, and (2) Block based algorithms. The first is not used for WMSNs because its heavy complexity load and huge associate size of data caused by the use of motion vector for each pixel. Whereas, the second class overcomes that by exploiting the high correlation between each pixel and its neighbors. It separately finds the motion vector of each block in the current frame according to matching process. Consequently, this class is called Block Matching Algorithms-BMAs [20,21]. A BMA searches for the best matched block , i.e. the lowest cost function among a group of candidate blocks posited in a search region that is called search window, which is limited by a search parameter-p. So the search will be performed within a square region of [-p,+p] around the position of the current block [21].

Each candidate block is seen as a Search Point-SP. Depending on the considered search points, BMAs are classified into two essential categories [20,22]: (1) Exhaustive Block Matching Algorithm-EBMA: It is called the full search algorithm because it examines all possible search points over the whole search window to choose the best matching one resulting in WMSN inappropriate load. (2) Fast Block Matching Algorithms-FBMAs: Unlike EBMA, the FBMAs try to reduce the complexity associated with EBMA [20,21]. Therefore, they examine not all but only some positions through the search window depending on a search pattern. As a result, FBMAs do not offer the same level of quality as EBMA, but they approximate it and moreover achieve a noticeable load reduction [20,22,23]. Accordingly, FBMAs are widely adopted by critical condition applications and constricted resources systems such as WMSNs [23]. In our view, FBMAs can be classified as shown in TABLE.I.

TABLE I
OUR BROAD CLASSIFICATION OF FAST BLOCK MATCHING ALGORITHMS.

| FBMA class | Characteristic | | |
|---|---|---|---|
| | ***Basic idea*** | ***Advantages*** | ***Disadvantages*** |
| Fixed search pattern based [12,13 ,14] | Search patterns with fixed and pre-determined shape and size, i.e. fixed templates. | Simplicity and regularity | Search inefficiency in tracking motion whose magnitude is not match the fixed search pattern size. |
| Inter-block correlation based [20,24,25] | Exploiting the coherency of frames blocks motion. | Decide a fit size of search pattern depending on the correlation between the current block and its neighboring blocks in the spatial and/or temporal domains | Additional resources for both: storing the neighboring MVs and computing the sutible search pattern size |
| Herarchical search based [26] | Exploring the correlation among different levels of representation of the same frame | Offering multi-resolution levels | High complexity and memory requirement not desirable for hardware implementation. |
| Sub-sampled pixels of blocks based [27] | Reducing the number of involved computation pixels by using only a fraction of pixels within a block | Speeding up the motion estimation and reducing the total computation | Additional cosiderations are needed to determine the block pixels reduction ratio to keep acceptable quality |
| Hybrid [28,29] | Fusing two or more ideas of the previous classes | Exploiting the advantages and avoid the disadvanteges | It depends |

### *E. Three Step Search Motion Estimation Algorithm*

As we already mentioned, authors in [10] applied TSS algorithm to predict motion vectors in motion estimation phase of the considered algorithm. We can conclude the mechanism of TSS as in Fig. 3. TSS under goes the fixed search pattern based FBMA class. The most important advantages of this algorithm are regularity and reducing the number of search points comparing with EBMA [20,21], where the maximum number of TSS search points (SP), i.e. the worst case is given as in (1) [22].

$$\max SP = 1 + 8\log_2(p+1) \qquad (1)$$

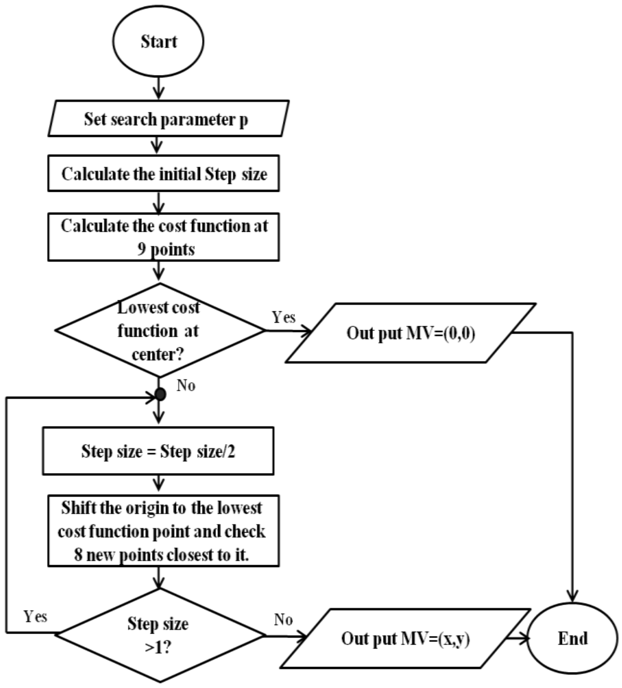


Fig. 3. Flow chart of three step search algorithm mechanism.

However, TSS suffers from search inefficiency when the size of the predetermined fixed search pattern does not match the magnitude of the actual motion in video [29]. So, if the motion magnitude is small comparing with the search pattern size, an over search will be incurred causing unnecessary searches and additional computational complexity which do not fit WMSN at all [20,22]. On the other hand, if the motion magnitude is large comparing (complex motion) with the search pattern size, an under search will be incurred yielding large matching errors and degrading video quality [22,29]. The above-mentioned observations and the basic issues of considered algorithm lay the foundation of our proposal to tackle the power consumption balancing idea.

## III. OUR ADAPTATION PROPOSAL

We did an accurate investigation about the most known FBMAs mentioned in TABLE I. Based on our researches, we believe that adopting an adaptive motion estimation search pattern instead of a fixed predetermined one is more efficient for WMSNs to avoid falling in over or under search problems. Therefore, we thought about Adaptive Rood Pattern Search-ARPS algorithm [29]. ARPS under goes the hybrid FBMA class, where we can consider it as a fusion of fixed search pattern class on a hand, and an inter-block correlation based class on the other. It greatly outperforms other FBMAs in terms of search accuracy, efficiency and computational complexity due to its distinct WMSNs fitted characteristics [22,29]. Unlike TSS, ARPS depends on the fact that the frame blocks motion is mostly coherent [29]. Thus, it uses a rood shaped pattern and exploits the motion vectors of the neighboring blocks to determine the rood arm length [20,29]. To save resources, ARPS depends only on the motion vector of the immediate left block of the current block which is called the predicted MV as in Fig. 4.

The most important advantage of this algorithm is that it depends on the adaptive pattern search and reduces the number of search points comparing with TSS, where the maximum number of ARPS search points, i.e. in the worst case, is given as in (2) [22].

$$\max SP = 1 + 4\log_2(p+1) \qquad (2)$$

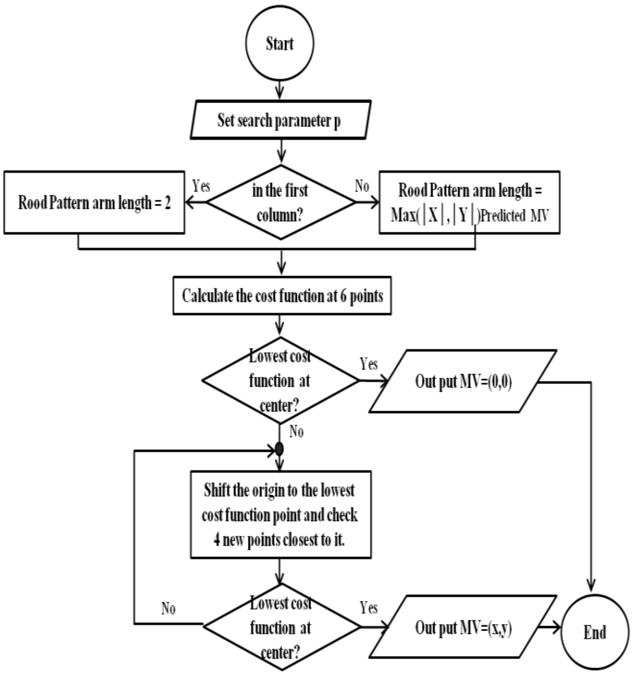


Fig. 4. Flow chart of adaptive rood pattern search algorithm mechanism.

ARPS inspirited us to modify the motion estimation phase of the edge feature-based video processing algorithm over WMSN by replacing the three-step search with the Adaptive Rood Pattern Search motion estimation algorithm in order to exploit its distinct characters which fit limited resources networks such as WMSN

## IV. RESULTS AND ANALYSIS

### A. .*Simulation Environment and Setup*

We used MATLAB 7.12.0 (R2011) to simulate the particularity steps of the considered algorithm. We ran the simulation on a DELL VOSTRO laptop computer with a 2.10 GHz Intel(R) Core(TM)2 Duo Microprocessor. Among various cost functions, i.e. block matching criteria, we use Mean Absolute difference-MAD because of its simplicity comparing with the other cost functions [23,21]. Two different videos are used : The first is Akiyo whose frames objects' motion changes slowly, and the second is Carphone whose frames objects' motion changes quickly. The stills of the analyzed videos are illustrated in Fig.5, and their characters are clarified in TABLE.II .

**Carphone** **Akiyo**

Fig. 5. Stills of the analyzed videos.

TABLE II
ANALYZED VIDEO CHARACTERS

| Analyzed Video | Characteristic | | | | |
|---|---|---|---|---|---|
| | ***Frame format*** | ***Frame size*** | ***Group of pictures*** | ***Number of frames*** | ***Motion mode*** |
| Akiyo video | QCIF[a] | 144x176 | 5 | 150 | Slow changes |
| Carphone video | QCIF | 144x167 | 5 | 150 | Quick changes |

[a] Quarter Common Interchange Frame.

### B. *Performance evaluation metrics*

In order to evaluate the performance of our proposal and measure its impact on the considered algorithm, we consider three basic evaluation metrics as following.

1. *Peak Signal to Noise Ratio(PSNR):* This metric measures the video quality. It is defined, for each video frame of size MxN, as in (3) [21].

$$PSNR = 10\log_{10}\left(\frac{(2^b-1)^2}{MSE}\right) \qquad (3)$$

Where b is the number of bits per pixel for the original frame, and MSE is the mean square error which is given as in (4).

$$\mathrm{MSE} = \frac{1}{\mathrm{M} \times \mathrm{N}} \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} \left[ x_1(i,j) - x_2(i,j) \right]^2 \quad (4)$$

Where x1(i, j) is the value of pixel (i, j) in the original frame, x2(i, j) is the value of pixel in the decompressed frame.

2. *Compression Ratio(CR):* It is a measure of the efficiency of compression algorithm in reducing both the required storage capacity and the data transmission power consumption, and it is given as in (5) [10].

$$CR = \frac{Size\ of\ Encoded\ Image}{Size\ of\ Original\ Image} \quad (5)$$

3. *Motion Estimation Power Consumption*: It is constant for a given search algorithm and different from one to another [30]. We did an extensive-deep investigation to find a direct model for the motion estimation power consumption. However, all the researches present it depending on aided or arbiter metrics which directly related to power consumption. For example: (1) [30] depended on measuring the current consumption in the case of coding/not coding program running, and then calculating the consumed power depending on the difference between the two cases, (2) other studies relied on average number of search points per block-either to directly represent the consumed power [23] or the execution time of the motion estimation algorithm [21], and (3) both [20] and [31] calculated the number of computational operations to represent the consumed power. Based on previous and on the importance of power consumption over WMSNs, we adopted the following metrics.

a) *Average number of search points per block:* It is the most common evaluation metric to compare among motion estimation algorithms. It is proportional with consumed power [21,23].

b) *Execution time of motion estimation algorithm:* We introduce the execution time as an actual time concept, it depends on the fact that the depression of algorithm execution time is an initial indicator to the depression of power consumption.

c) *Complexity degree of motion estimation algorithm:* The less execution time required algorithm is not necessarily the simpler one [32]. According to that fact and avoiding to depend only on execution time to determine the algorithm complexity, we investigated the complexity order, i.e. the big O notation [33], of both TSS and ARPS algorithms. To the best of our knowledge, analyzing the big O complexity degree of these algorithms has not been considered in the literature. When we try to find the complexity of the algorithm, we do not interest in the exact number of operations that are performed. Instead, we interest in the relation of the number of operations to the algorithm related important factor which is called input size or the problem size usually mentioned as n [33]. Typically, we usually care about the worst case, i.e. what the maximum number of operations that might be performed for a given problem size is? [33].

d) *Computational complexity:* This measure is used to decide which one of equal comlexity order algorithms is the simplest [20,31]. Where, we depend on the number of basic operations an algorithm performs, and the less number of basic operations algorithm is the less complexity and power consumption.

### C. *Simulation results*

To evaluate the efficiency of our proposed adaptation, several extensive simulations are conducted. We implemented the edge-feature based video processing algorithm in two cases: as proposed in [10], i.e. TSS based motion estimation, and after applying our adaptation proposal, i.e. ARPS algorithm based motion estimation. We examined the two videos, Akiyo and Carphone, and got the following results for the performance evaluation metrics.

1) *Peak Signal to Noise Ratio:* Simulation results for PSNR of each P-frame of Akiyo and Carphone are shown in Fig. 6 and Fig. 7, respectively. We can notice that our proposal of using ARPS maintains the video quality because its PSNR results compete the results of the considered algorithm which uses TSS. ARPS does not cause a degradation of the PSNR values of both videos. Where, the average PSNR values for ARPS and TSS are 36.22dB and 36.25dB respectively for Akiyo video, and 33.13dB and 33.19dB respectively for Carphone video. And in each video, results are with less than 0.45 dB difference and with a very small value for the mean square difference by 0.02.

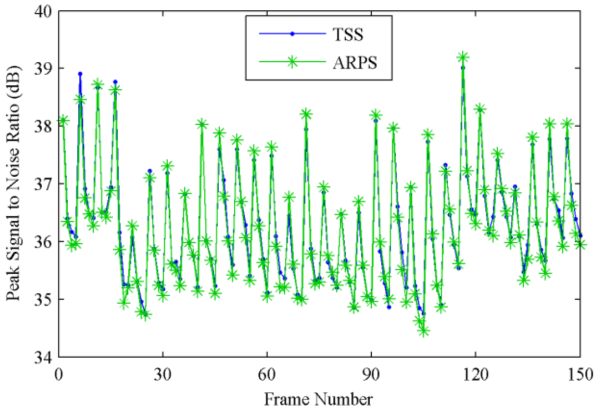


Fig. 6. Peak Signal to Noise Ratio, Akiyo video.

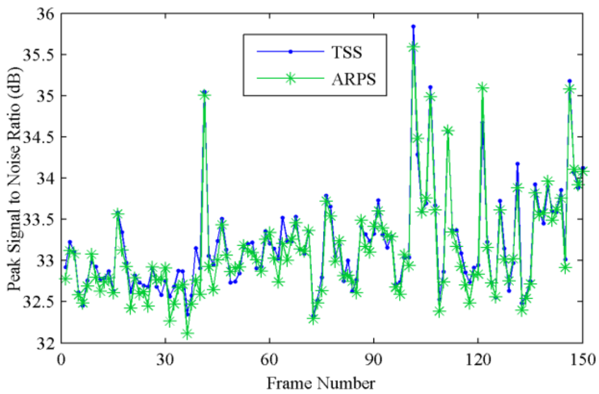


Fig. 7. Peak Signal to Noise Ratio, Carphone video.

2) *Compression Ratio (CR):* We plot the P-frame-by-P-frame compression ratio results for ARPS and TSS for both Akiyo and Carphone videos in Fig. 8 and Fig. 9, respectively.

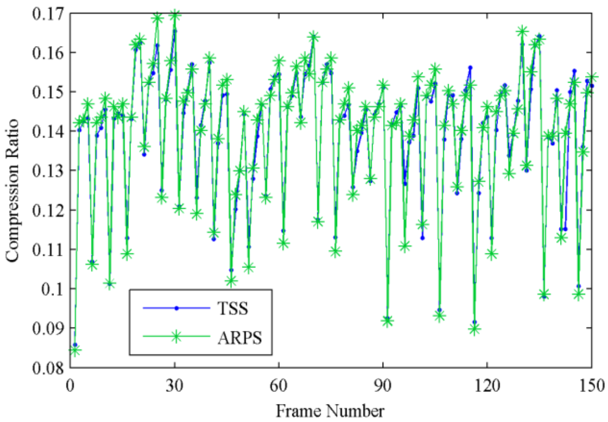


Fig. 8. Compression Ratio, Akiyo video.

It is obvious that ARPS saves the frames compression ratio and does not require neither additional memory nor additional power for transmitting the compressed video data. Where, the average CR values for ARPS and TSS are 0.1394 and 0.1389 respectively for Akiyo video, and 0.1987 and 0.1977 respectively for Carphone video. And in each video, the results are with less than 0.025 difference and with a very small value for the mean square difference by 1.3 e-5.

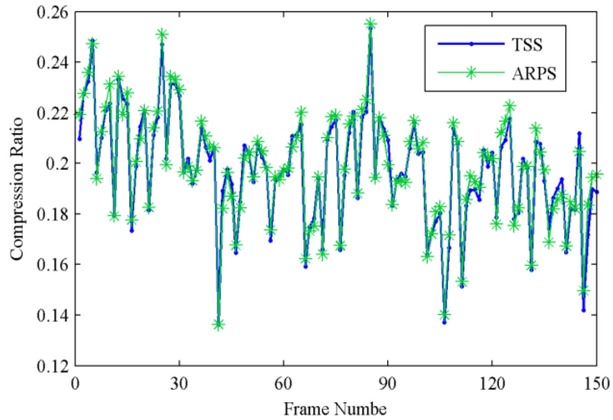


Fig. 9. Compression Ratio, Carphone video.

3) *Motion Estimation Power Consumption:* We adopted the following metrics to evaluate the power consumption.

a- *Average number of search points per block*: From Fig. 10 and Fig. 11, which illustrate the results of average number of search points per block of P-frames, we can clearly notice the excellent performance of our proposal of adopting ARPS. It requires less average number of search points per block comparing with TSS at all frames and for the both analyzed videos. The average percentage reduction rates are 56% for Akiyo video, and 47% for Carphone. Due to the proportionality of this metric with consumed power, we can say that this proposal saves the power obviously. Moreover, we can notice that TSS requires pretty close values of average number of search points per block from frame to another comparing with ARPS which requires fluctuating values. That is because TSS adopts a fixed-predetermined search pattern, while ARPS adopts an adaptive search pattern which tracks the motion and consequently checks difference numbers of search points per block according to the motion magnitude. We calculated the standard deviation statistical parameter for average number of search points per block which asserts our last notice. Its values for average number of search points per block for ARPS and TSS are 0.72 and 0.06 respectively for Akiyo video, and 0.09 and 0.82 respectively for Carphone video.

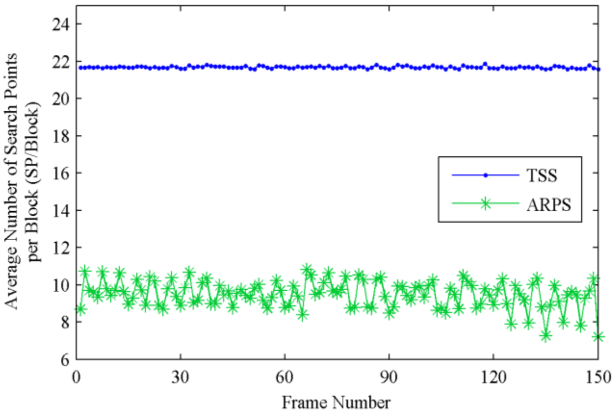


Fig. 10. Average number of search points per block, Akiyo video.

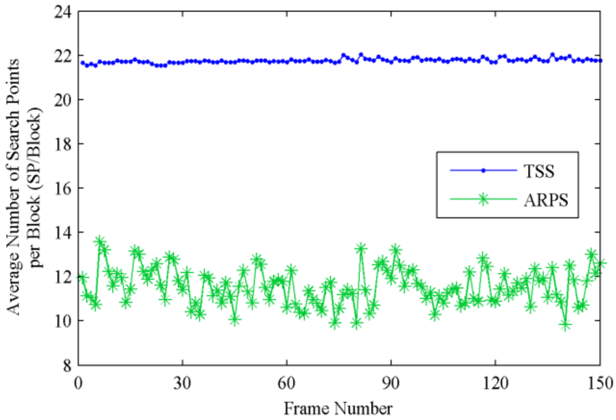


Fig. 11. Average number of search points per block, Carphone video.

b- *Execution time of motion estimation algorithm:* In Fig. 12 and Fig. 13, we measured the Execution time of TSS and ARPS for Akiyo and Carphone respectively. The results again show the excellent performance of our proposal of adopting ARPS. It achieves less excutive time comparing with TSS at all frames and for the both analyzed videos. The average percentage reduction rates are 56% for Akiyo video, and 47% for Carphone. Another point, we can also notice that TSS requires pretty close values of execution time from frame to another comparing with ARPS which requires fluctuating values. The standard deviation for execution time asserts that. Where, its values for execution time for ARPS and TSS are 0.26 and 0.05 respectively for Akiyo video, and 0.06 and 0.29 respectively for Carphone video.

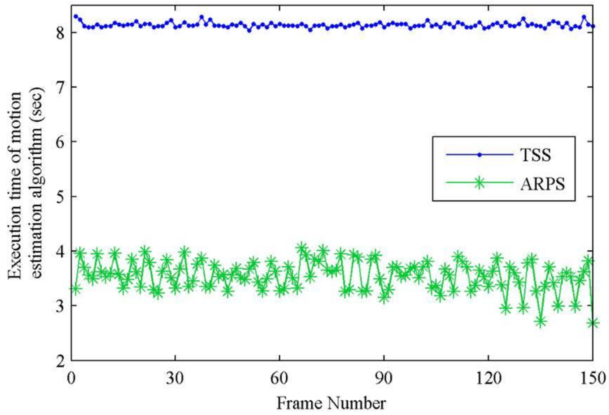


Fig. 12. Execution time of motion estimation algorithm, Akiyo video.

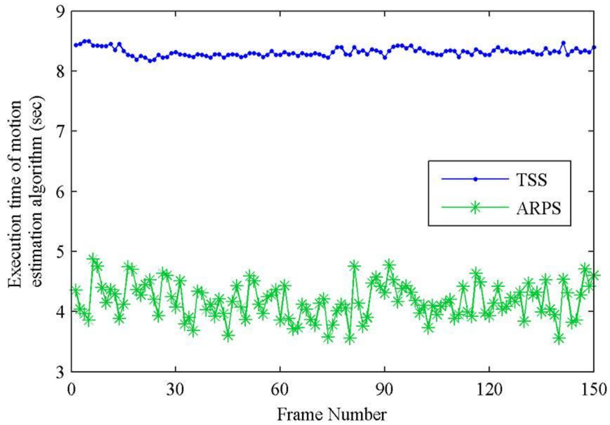


Fig. 13. Execution time of motion estimation algorithm, Carphone video.

c- *Complexity degree of motion estimation algorithm*: To the best of our knowledge, analyzing the big O complexity degree of TSS and ARPS algorithms has not been considered in the literature. However, to detect the effect of our proposal on the comlexity of the considered algorithm, we analyzed the complexity degree of the two motion estimation algorithms depending on their mechanismes and the worst case of search points which is clarified in (1) and (2). Search parameter p represents the proplem size. According to our complexity analysis, we figure out that ARPS and TSS have the same degree of complexity which is the logarithmic degree of the problem size, i.e. O(log(n)). This result puts forward an initial pointer to the neutralization of the complexity of ARPS and TSS. As a result, our proposal does not cause additional complexity to the algorithm

d- *Computational complexity:* When determining the complexity degree of an algorithm, we do not interest in the exact number of operations that are performed. But, because of the WMSN characteristics and its sensitive constraint resources we relied on the exact numbers of computations of TSS and ARPS to improve the simplicity of our proposal. Fig. 14 and Fig. 15 show the diagrams of the number of computations resulting in motion estimation process for ARPS and TSS, and for Akiyo and Carphone respectively. It is clear that ARPS is simpler and thus more suitable to be adopted for WMSN applications.

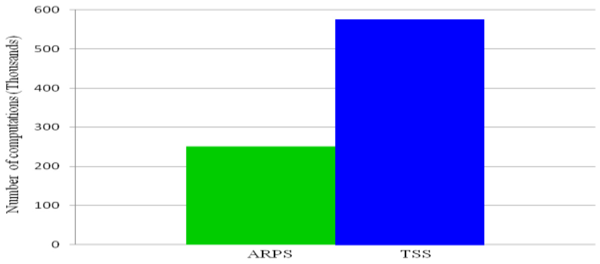


Fig. 14. Computational complexity, Akiyo video.

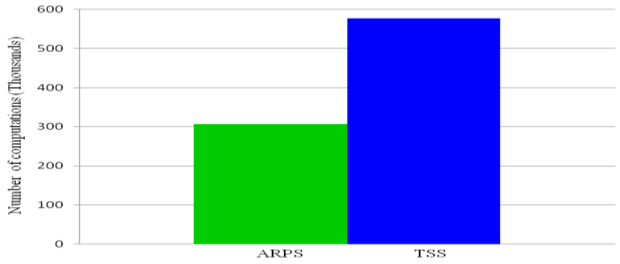


Fig. 15. Computational complexity, Carphone video

## V. CONCLUSION AND FUTURE WORK

In this paper we have adopted a motion estimation algorithm which reduces the total number of search points for finding the motion vector in the search window comparing with the considered algorithm, and then improve video coding efficiency by reducing temporal redundancy existing in a video sequence which is more efficient for WMSNs. We also introduced a new method of power consumption to compare among the different block matching algorithms which is very important issue for WMSN.

As a future work, we plan to exploit the energy reduction of the adopted proposal to develop the used processing techniques in order to improve both video quality and compression ratio, such as the adoption of more efficient coding standards.

## REFERENCES

[1] T.N Prabhu, C. Ranjeeth Kumar, and B. Mohankumar, "Energy-efficient and Secured Data Gathering in Wireless Multimedia Sensor Networks", International Journal of Innovative Research in Computer and Communication Engineering.Vol. 2, Issue 2, p.p. 3073-3079, 2014.

[2] S. Atif, P. Vidyasagar, and C. Elizabeth, "Wireless Multimedia Sensor Network Technology:A Survey". 7th IEEE International Conference on Industrial Informatics (INDIN) IEEE.p.p.606-613, Cardiff, Wales, 2009.

[3] R. Denis, M. Kassio, A. Antonio, M. Dionne, and C. Eduardo,"Recent Advances And Challenges In Wireless Multimedia Sensor Networks",Chapter 4 of Book: Mobile Multimedia-User and Technology Perspectives, Publisher InTech, ISBN:978-953-307-908-0, 2012.

[4] L. Sung-Keun, K. Jin-Gwang, and J. Chang-Ryul, " An Energy-Efficient QoS-aware Routing Algorithm for Wireless Multimedia Sensor Networks", International Journal of Multimedia and Ubiquitous Engineering Vol.9, No.2, p.p.245-252, 2014.

[5] A. Junaid Jameel, K. Hassan Aqeel, and K. Syed Ali, " Energy Efficient Video Compression For Wireless Sensor Networks ". in Proceedings of the 43rd Annual Conference on Information Sciences and Systems (CISS), p.p. 629–634, Baltimore, MD, USA, 2009.

[6] M. Abdelhamid., H. Brahim, and K. Ahmed," A Survey Of Image Compression Algorithms For Visual Sensor Networks ". International Scholarly Research Network (ISRN) Sensor Networks, vol. 2012, Article ID 760320, p.p.1-19, doi:10.5402/2012/760320,(2012.

[7] R. Y. Sree, and R. N. Kamesh,"Image Change Detection Using Wireless Sensor Networks", Proceedings of the 3rd IEEE international conference on Distributed computing in sensor systems, (DCOSS'07), PP: 240-252, New Mexico, USA, 2007.

[8] D. Frederic, G. Wen, T. Stefano, and V. Anthony," Distributed Video Coding: Trends And Perspectives", EURASIP Journal on Image and Video Processing, (ISSN: 1687-5176), Vol. 2009 , Hindawi Publishing Corporation, 2009.

[9] C. Yui-Liang, and Y. Hong-Hsu, "MAC-Aware and Power-Aware Image Aggregation Scheme in Wireless Visual Sensor Networks". Journal of Sensors.vol.2013, p.p.1-13, 2013.

[10] H. Phat Nguyen, Q. Vinh Tran, and M. Takumi, "Video Compression Schemes Using Edge Feature On Wireless Video Sensor Networks ", Journal of Electrical and Computer Engineering, vol. 2012, Article ID 421307, p.p.1-20,(2012.

[11] S. Shubhashree, "A Review on Edge Detection Techniques for Image Segmentation", International Journal of Computer Science and Information Technologies (IJCSIT), Vol. 5, No.4, p.p. 5898-5900, 2014.

[12] D. Soumya, and B. CH. Bidyut, "A Statistics and Local Homogeneity based Color Edge Detection Algorithm", IEEE International Conference on Advances in Recent Technologies in Communication and Computing, p.p. 546-548, 2009.

[13] B. Sriram, M. Eswar Reddy, and G. Subha Varier," Study of various motion estimation algorithms for video compression/coding standards & implementation of an optimal algorithm in LabVIEW", International Journal of Emerging Technology and Advanced Engineering, Volume 3, Issue 4,p.p. 372-381, 2013.

[14] T. Koga, K. Inuma, A. Hirano, Y. Iijima, and T. Ishiguro, "Motion-Compensated Interframe Coding for Video Conferencing", Proceedings National Telecommunications Conference, New Orleans, '81 (IEEE), p.p. 531-535, 1981.

[15] P. Davis, and M. Sangeetha, "Implementation of Motion Estimation Algorithm for H.265/HEVC", International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering, Vol. 3, Special Issue 3, p.p.122-126, 2014.

[16] B. Francesca, K. Michel, and W. Claudio, " Wyner-Ziv Coding With Uncertain Side Information Quality ", 18th European Signal Processing Conference (EUSIPCO-2010), pp.2141-2145, Denmark 2010.

[17] M. Abdelhamid, K. Ahmed, Z. Djemel, and H. Brahim, " Energyaware Jpeg For Visual Sensor Networks ". in Proceedings of the MCSEAI Conference, pp. 639–647, 2008.

[18] A. sh. Muhammad, and S. B. Sagar, " Video Compression Algorithm Using Motion Compensation Technique: A Survey", International Journal of Advance Research in Computer Science and Management Studies, Volume 2, Issue 3,p.p. 357-363, 2014.

[19] K. mohammad reza, "Evaluation of different block matching algorithms to motion estimation", International Journal of VLSI and Embedded Systems-IJVES, Vol 03, Issue 03, p.p.148-153, August 2012.

[20] M. Kaushik,"Comparative Analysis of Exhaustive Search Algorithm with ARPS Algorithm for Motion Estimation", International Journal of Applied Information Systems (IJAIS),Vol.1, No.6, p.p.16-19, 2012.

[21] Y. Razali, A. Alihossein, A. H. Alfian, and S. Nasir, "A Comparison of Different Block Matching Algorithms for Motion Estimation", ScienceDirect, The 4th International Conference on Electrical Engineering and Informatics –ICEEI, p.p. 199 – 205, Malaysia, 2013.

[22] L. C. Manikandan, and R. K. Selvakumar, " A New Survey on Block Matching Algorithms in Video Coding", International Journal of Engineering Research (IJER), Vol.3, Issue No.2, p.p. 121-125, 2014.

[23] Ch. Cheong, B. J. Asral, and H. Razaidi, "Review of Energy Efficient Block-Matching Motion Estimation Algorithms for Wireless Video Sensor Networks", IEEE Symposium on Computers & Informatics, p.p. 241-246, 2012.

[24] H. Nisar, and Ch. Tae-Sun, "An adaptive block motion estimation algorithm based on spatio-temporal correlation", IEEE International Conference on Consumer Electronics ICCE, pp.393 – 394, Las Vegas, USA , 2006.

[25] C. H. Hsieh, P. C. Lu, J. S. Shyn, and E. H. Lu, “Motion Estimation Algorithm Using Interblock Correlation,” Institution of Engineering and Technology IET, Electron. Lett., vol. 26, no. 5,p.p. 276–277, 1990.

[26] A. Heinrich, C. Bartels, R. J. van der Vleuten, C. N. Cordes, and G. de Haan,"Optimization of Hierarchical 3DRS Motion Estimators for Picture Rate Conversion", IEEE Journal of Selected Topics in Signal Processing, Vol 5, no.2, pp. 262-274, 2011.

[27] G. B. Saverio, and I. Ebroul, “Residual error curvature estimation and adaptive classification for selective sub-pel precision motion estimation”, IEEE

International Conference in Acoustics Speech and Signal Processing (ICASSP), p.p. 1193–1196, 2012.

[28] N. K. Piyush, and R. B. Dhaval, "A New Hybrid Block Based Motion Estimation Algorithm for Video Compression", International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering, Vol. 3, Issue 5,p.p. 9586- 9596, 2014.

[29] N. Yao, and Ma. Kai-K, "Adaptive Rood Pattern Search For Fast Block-Matching Motion Estimation", IEEE Transactions on Image Processing, Vol.11, No.12, p.p. 1442–1449, 2002.

[30] L. Xiaoan, E. Elza, W. Yao, and G. David, " Power Efficient Multimedia Communication Over Wireless Channels ", IEEE Journal On Selected Areas In Communications, Vol. 21, No. 10, p.p. 1738-1751, 2003.

[31] H. T. Trung, Ch. Hyo-Moon, and Ch. Sang-Bock, "Performance Enhancement of Motion Estimation Using SSE2 Technology ", World Academy of Science, Engineering and Technology Vol:2, No4, p.p. 161-164, 2008.

[32] IngoWegener, "Complexity Theory, Exploring the Limits of Efficient Algorithms", Springer-verlag Berlin, 2005.

[33] Goldreich, Oded. "Computational Complexity: A Conceptual Perspective". Cambridge University Press. ISBN 978-0-521-88473-0, 2010.