A Compression Algorithm for Video Surveillance System

Dhafer R. Zaghar
Asst. Prof., Computer and Software Dept. Engineering College
AL-Mustansiryah University

Dhulfiaqr S Aldeen A Alwahab
Asst. Lect., Computer and Software Dept. Engineering College
AL-Mustansiryah University

Tawfeeq Enad Abdulabbas
Asst. Lect., Computer and Software Dept.Engineering College
AL-Mustansiryah University

ABSTRACT
The major challenge in surveillance system and visual sensor network lies in how to compress and store video. The deployment from a Digital Video Recorder (DVR) to Network Video Recorder (NVR) made the challenge more complicated, since there is no central unit controlling on the compression and highlight the big data problem on the other hand. This paper presents an algorithm to compress videos considering the contained information. Information is the result of a combination between faces and motion detection techniques. That is when information increases the quality of the stored frame (image) increase and vice-versa. The proposed algorithm is tested under different videos with varied time and information. These videos are also compressed using JPEG-2000 algorithm. The obtained results showed that the proposed algorithm produced less video size with more detailed information than the others.

General Terms

Surveillance storing algorithm, JPEG-2000 (frame-by-frame) algorithm, Face detection, Information and motion estimation

Keywords
Monitoring systems, Video compression, (CRR) Compression Reduction Ratio.

1. INTRODUCTION
Security system had been considered as a vital and important topic in the peoples' lives. It helps them to maintain their personal possessions belongings and helps to warn them before the disaster occurred as in alarm sensory devices; it may provide them with visual evidence about what happened before or during the occurrence of crimes. The traditional alarm systems that depend mainly on the sensors like fire alarm, gas leak alarm, have a high probability of false alarm rate and can easily trigger to give a fake alarm. Also, sensor alarm cannot maintain a remote area in real time to reduce losses [1]. Vision based camera systems were adopted in security systems to construct monitoring systems based on images and videos [2]. These systems can provide people with accurate and truthful information as compared with sensor systems that are limited in measure or quantified or in solving a specific problem [3].

2. PROBLEM FORMULATION
Security monitoring system usually uses a large number of devices to provide obvious, clear and accurate information. An increasing number of devices, create large amounts of data that cause a challenge in processing which include capture, storage, transfer, analysis and visualization. Monitoring has become a big data problem [4], some previous works deal with big data by using data management as in [5]. The main purpose of the monitoring system is providing visual information about an event happen in critical time or during a small period of time.

Visual information should be very clear and available (stored). Whereas most time the monitoring device exhausts resource by keeping an eye (camera) up for empty places or watching a normal activity providing large amounts of data without any important information. Actually, these redundant data made the monitoring system stand up to big data problem and need more expensive devices with high storage capacity, processing time and power consumption. What the monitoring system need is a way to extract information from the big data. The monitoring system is considered as a distributed system structure, based on, camera and image to collect data, then image processing techniques could be used to extract information. Different compression techniques are proposed in the literature for controlling and reducing video size on the cost of losing information and details. But, the problem is how to controlling video size and keeping information and details available and clear as much as possible.

3. SURVEILLANCE STORING ALGORITHM
Generally, in the distributed system, estimating the amount of information in images could be used to decide how the image can be stored through the benefits obtained by using frame-by-frame compression technology [6]. The aim of this work is to introduce a surveillance algorithm that can decide what to monitor, and what information must be captured that made the overall system efficient that can be able to handle the monitoring goals with high quality and can reduce the big data based on the information in the image.

The proposed algorithm is designed to store different quality levels for the captured images, depending on the type of the required information and on the levels of the details in the scene. Thus, the stored image quality will be directly proportional to the demand details of the information.

In this algorithm, the amount of information to be stored in an image depends mainly on two features: human face and the movement of an object in the captured scene. Extracting the information about these features can be done by using the two basic well known techniques, face detection [7, 8] and motion estimation. Information that is obtained from the image (monitoring device) will control of the image quality by direct proportion (high information gives high image quality and priority).

Information estimation based on two basic well known techniques, face detection and motion estimation, the combination of these two image processing algorithms represent the amount of the information and as a consequence the way of image storage. As shown in the "Fig. 1,"

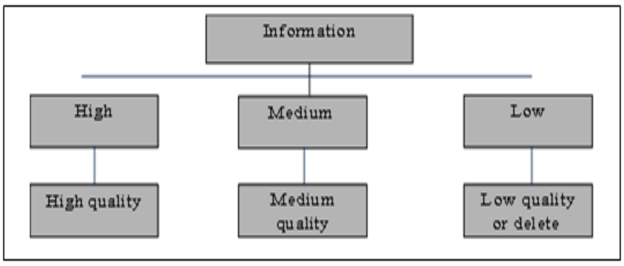

Fig. 1. Relation of Information to Image Quality

When the monitor device detect face, the information trigger to high that give indication to store the image with high quality and capture the next image to check if there is more new faces, as there is a new face the information indication remain high and image storage quality remain high too, when the monitoring device detect no more faces the motion detection method is used to observe (record) the movement of these faces, when it detect a motion, the information indication trigger to medium to store the image with less resolution, while if there is no motion, the information indication become low and image storage set to low. "Fig.2," shows a flowchart of the proposed algorithm.

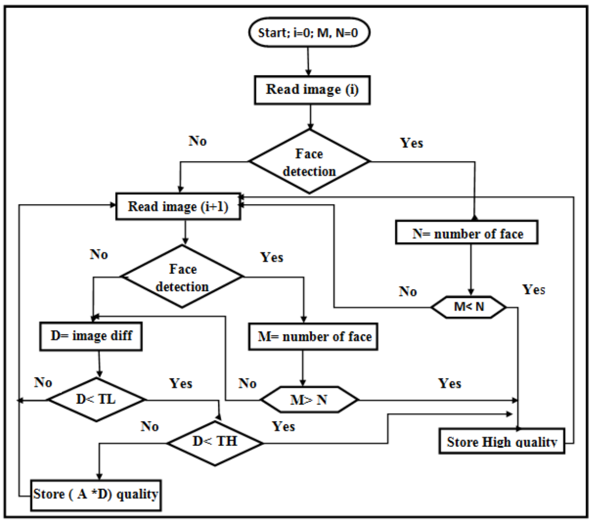

Fig. 2. Proposed Algorithm Flowchart

4 . VIDEO COMPRESSION TECHNIQUES

The video file size could be reduced in various techniques to be more suitable to store or transmit without loss of full information (keeping the suitable amount of data). These techniques are mostly known as image compression. Many algorithms are available and used for compression, some are lossless by keeping the same information as the original image, some others loss information when compressing the image. Compression technologies in video surveillance systems can categorize into; frame-by-frame compression and temporal compression [9]. Among them Joint Photographic Expert Group (JPEG) which is a well-known (frame-by-frame) algorithm, one of the characteristic that make JPEG very flexible is the compression rate that can be adjusted permitting the user controlling how much loss of information to introduce a trade-off between file size and quality. For the proposed work JPEG-2000 will be used based on the result in [9] as well as it's designed to preserve as much detail and evidence as possible within the image while greatly reducing file sizes. Also JPEG- 2000 was designed to preserve the extra information that the sensors generate and maintain it in the compressed video [6]. The amount of JPEG compression is measured in a percentage of the quality level, an image at 100% quality has (almost) no loss, and 1% quality is very low quality, or some time in compression rate, more compression ratio yield lower quality and smaller size.

5. ALGORITHM IMPLEMENTATION

The proposed algorithm had been implemented using MATLAB. Many algorithms were proposed in the literature for face detection. Viola and Jones algorithm [6] demonstrated that faces can be fairly reliably detected in real-time (i.e., more than 15 frames per second on 320 by 240 images with desktop computers) under partial occlusion, for that reason it had been adopted in this work. Whereas, in the presence of the new face detection, basic two successive frame subtraction was carried out to estimate motion detection. The proposed algorithm assumed two threshold values TL and TH which stands for Low threshold and High threshold respectively. According to the relation illustrate in "Fig. 3," if the difference value (D) between two successive frames is less than TL, then the image of the second frame will not store. On the other hand, if (D) is greater than TH, then the quality of the image to be stored will be 100%. Otherwise the relation between the difference value (D) and the quality of the image is linear (i.e. for the period between TL and TH). For all videos, regardless of the number of faces or values of (D), first frame (image) is stored with roughly the same quality of JPEG-2000.

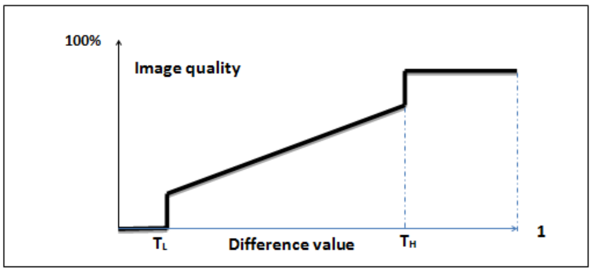

Fig. 3.Relation between Frame Difference and Image Quality

The difference value (D) in two successive frames is the Mean value to the absolute result obtain by subtract the current frame from the preceding one. The maximum unreachable value for (D) is (255) which resulted by subtract a white from black image and Mean the result matrix. Normalizing the value of (D) to (1) is achieved by using this maximum value (255). Image quality (Iq) can be defined by

$$Iq = \begin{cases} 0, & D < TL \\ a * D, & TL < D < TH \\ 100\%, & D > TH \end{cases} \qquad (1)$$

Several tests have been made to select the effective thresholds and (a) values. In these tests different faces views with different sites and situation have been considered. It is found that, the effective (TL) is equal to (0.01960) and (TH) is equal to (0.0784). Whereas (a) is a constant scalar value suggested being (1.5).

6. EXPERIMENTAL RESULTS

6.1 Test 1: General Test

To test the proposed compression algorithm, an off line video had been captured using the image Acquisition Toolbox supported in MATLAB. An RGB24 uncompressed video format (AVI) has (33) frame (9) fps, height (768) and width

(1024) in (3.66) Sec produce a file with (76,098) MB in size. The main compression technologies currently used in video surveillance JPEG-2000 are used to convert the recorded video into a stream of images. Independently, the propose algorithm is also used to convert the recorded video to another stream of images. To highlight the proposed algorithm, the reader should be informed that information in the video is summarized by a single person (one face), sitting in front of the camera, moving slowly (Spasmodic movements). The result of the two conversion process is shown in "Fig. 4," where X and Y axis, represent frame number and frame size in KB, respectively. The images produced from the JPEG-2000 process, roughly have the same size in the range (69-74 KB) with the same quality for each image (frame). On the other hand, the proposed algorithm produces images varying in size based on the important information that's contained in the frame (face and motion). The first frame, JPEG-2000 stored it as any frame in the video regardless on the information. Whereas the proposed algorithm stores it with a full quality (100%) since there is information about a new face. In the second frame, the proposed algorithm doesn't record any variation in the number of faces nor obvious movement, the image was stored with a very low quality that cause a reduction in their size.

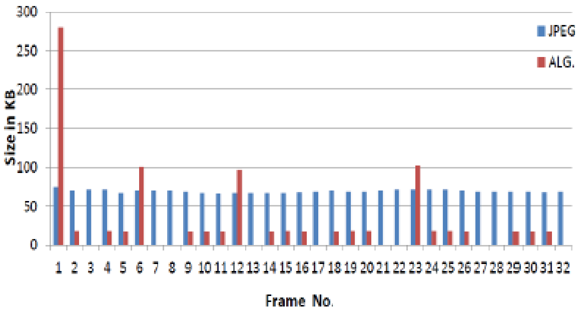

Fig.4. ALG and JPEG-2000 Frame Compression

For the third frame, the proposed algorithm doesn't detect new face with a value of (D) less than (TL), which cause frame neglecting. For frame six, the proposed algorithm detects a motion by the person that made frame storing with quality larger than the preceding frame, and so on. Another video was also recorded using the same camera properties to monitor different activity (person sitting in the monitor area then move away. After a while, another one came to appear). Also, another test was done by recoded a third video using the same camera to monitor an empty area (no information), the results of this test shows that the ratio of the total size between the proposed algorithm and the classic JPEG-2000 is about 39% as in row 1 in table 1. This test repeated for two different films for real monitoring cameras used in a computer department, first for the main entry while the second for the lane in the third floor. The proposed algorithm reduces the total size to less than 13% and less than 1% respectively as shown in the table 1.

Table 1 Video compression size results

Test No.	Video Size (AVI)MB	No. of frames	JPEG-2000 MB	P.Algorithm MB	P.Algorithm to JPEG size reduction ratio
1	76,098	33	2,284	0,891	39.01%
2	359,497	156	7,710	0,993	12.87%
3	156,739	68	3,230	0.018	0.55%

6.2 Test 2: Specific Test

The results in table.1 show that the proposed algorithm is a useful approach which can reduce the total size of the compressed films from less than 40% (test 1) to less than 1% (test 3). This leads that the proposed algorithm is useful, but for more advantages it still needs a test (even for specific cases) to make the algorithm be more beneficial.

For security (monitoring) systems camera usually records different type of activity. Activity time varied from high information, during working peak time, medium, low or no information. For example, security monitoring cameras in an office record/transmit video at the same quality as long as it is on. In normal situation, this office has the most information during working time, least and important information at night (or off time). Four different un-compressed video (AVI) had been recorded during different times in an office using the same camera. First video, persons (faces) moving normally in the area. Second, the object (without force) moving normally. Third, object moving un-normally and fast in random direction. Fourth, no face and no object movement (during mid-night). The proposed algorithm and JPEG-2000 are also used to compress these four videos, table 2 shows the size reduction in each one as well as the proposed algorithm to JPEG-2000 Reduction Ratio (RR).

Table 2 Video size reduction

NO.	Video size (AVI) KB	JPEG-2000 KB	P.A KB	RR	Information
1	259,274	5710	1890	0.33	Faces with normal motion
2	185,471	4200	817	0.19	Object without face moving normally
3	241,273	5930	1640	0.27	Object without face moving fast and un-normally
4	156,670	3090	20	0.006	Empty area (no information)

6.3 Test 3: Daily Working Test

The results in table 2 show that the proposed algorithm is better than JPEG-2000 in all cases, but the total effect of this algorithm need to test it for real case in a daily working.

The overall Compression Reduction Ratio (**CRR**) for these four video activities, is computed using simple weighted sum equation (2).

$$\mathbf{CRR} = \sum \boldsymbol{W.RR} \quad (2)$$

W is the percentage weight of the activity hours to the 24 hours. RR is the size reduction of the proposed algorithm to JPEG-2000 Reduction Ratio as is seen in table 2.

Considering the camera is installed in an office, a weight (W) is suggested for each activity. Daily working office time was assumed from (7:00 AM – 2:00 PM), the video for this activity is assumed in the first recorded video. The office will close from (2:00 PM to 7:00 AM next day), this activity represents as in fourth video. Whereas the second and third video could happen during any time in the day.

For first video it occupies 29% from the overall day and 62% for the last video, while for Second and third video (W) had been suggested to be 5% and 4%, respectively. In this case steady the CRR is:

CRR= [0.29(0.33) + 0.05(0.19) + 0.04(0.27) + 0.62(0.006)]= 11.822

Then the overall compression reduction of the proposed algorithm to the JPEG-2000 is about 12% for this test. More daily tests for different cameras and different offices show the value of CRR (overall compression reduction) become full in the range 7% to 22% as a maximum, while the average value of CRR is 14.321%.

7. CONCLUSIONS

This paper presents a video compression algorithm for monitoring systems based on the widely used standard frame-by-frame JPEG-2000. The algorithm architecture is organized using two well-known techniques for estimating the amount of information, in each captured image; face detection and motion detection. The algorithm deduces the new captured image has more information when it detects faces or discovers a difference than the previous one. Finally, image stored quality affected directly by the amount of acquired information. The tests results in (table 1) show that the proposed algorithm reduces the size of all videos while preserving the amount of information gained from the monitored area as much as possible. Videos used in tests contain different monitored activity and varied length time, which could be obtained by dividing the number of frames in each video from (table 1) by (9) fps. The higher size reduction appearing in test (3), where the camera was monitoring area with no activity (no information).

In addition to the reducing in video size, more details (zooming) could be obtained from images stored in high quality than JPEG-2000.

The proposed algorithm reduces the size of the compressed films in a good ratio (about 14.321%), that could be considered a type of help to solve the big data problem in distributed systems and visual sensor network. So the proposed algorithm increases the quality of the important information of the monitoring systems (face and motion).

The proposed algorithm will add a new process to the compression algorithm that will need more processing time or additional hardware. The additional hardware (processing function) can be satisfied using FPGA or using special image processing hardware, this will add more cost, but the proposed algorithm will reduce the total size to less than 15% from the JPEG-2000, in result the total cost for the system will become cheaper.

8. REFERENCES

[1] Mic Bowman , Saumya K. Debray , Larry L. Peterson, Reasoning about naming systems, ACM Transactions on Programming Languages and Systems (TOPLAS), v.15 n.5,p.795-825,Nov.1993 doi:10.1145/161468.161471].

[2] Ding, W. and Marchionini, G. 1997 A Study on Video Browsing Strategies. Technical Report. University of Maryland at College Park.

[3] Fröhlich, B. and Plate, J. 2000. The cubic mouse: a new device for three-dimensional input. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems

[4] Tavel, P. 2007 Modeling and Simulation Design. AK Peters Ltd.

[5] Sannella, M. J. 1994 Constraint Satisfaction and Debugging for Interactive User Interfaces. Doctoral Thesis. UMI Order Number: UMI Order No. GAX95-09398., University of Washington.

[6] Forman, G. 2003. An extensive empirical study of feature selection metrics for text classification. J. Mach. Learn. Res. 3 (Mar. 2003), 1289-1305.

[7] Brown, L. D., Hua, H., and Gao, C. 2003. A widget framework for augmented interaction in SCAPE.

[8] Y.T. Yu, M.F. Lau, "A comparison of MC/DC, MUMCUT and several other coverage criteria for logical decisions", Journal of Systems and Software, 2005, in press.

[9] Spector, A. Z. 1989. Achieving application requirements. In Distributed Systems, S. Mullender, Ed. ACM Press Frontier Series. ACM, New York, NY, 19-33. DOI= http://doi.acm.org/10.1145/90417.90738.