

A Review on Design and Implementation of Compensated Frame Prediction & Reconstruction

Neha¹, Ms. Amrita Chaudhary²

M.Tech Scholar, Computer Science & Engineering Dept, Galaxy Global Group of Institutes, Kurukshatra¹

Asst. Professor, Computer Science & Engineering Dept, Galaxy Global Group of Institutes, Kurukshatra²

Abstract: This paper presents a review detecting and extracting each frame in video sequence for detecting motion between these frames. In image and video processing, the estimation of motion plays a vital role in video compression as well as multi-frame image enhancement. Disparate as they may seem, these many applications share one common thread in all such applications, the demand is high for accurate estimates of motion requiring minimal computational cost. In this, a new technique called Motion Detection using Pixel Processing is proposed. The proposed technique attempts to decrease the computation time between the current macro-block and the macro-blocks in the reference frame, by stopping the calculation of the sum absolute different between the pixels in the current and the reference frame macro-blocks when the current uncompleted sum absolute value is greater than the previous calculated one. This will use MATLAB tool for simulation.

Keywords: Motion Estimation, background Subtraction, Block Based Method, Pixel Based Method etc.

I. INTRODUCTION

In equivalent with space requests, image processing methods began in the late 1960s and early 1970s to be used in medicinal imaging, remote Earth possessions observations, and astrophysics. The invention in the early 1970s of computerized axial tomography (CAT), also called high-tech tomography for short, is one of the most significant events in the presentation of image processing in medical analysis. Computerized axial tomography is a process in which a ring of indicators encloses an object (or patient) and an X-ray source, concentric with indicator ring, rotates about the article. The X-rays pass through the entity and are collected at the opposite end by equivalent detectors in the ring. As the source swaps, this procedure is repeated. Tomography consists of algorithms that use the detected data to construct an image that represents a “slice” through the entity.

Motion of the object in a direction upright to the ring of detectors yields a set of such slices, which establish a three-dimensional (3-D) version of the inside of the entity. Tomography was conceived independently by Sir Godfrey Hounsfield and Professor Allan Cormack, who joint the 1979 Nobel Prize in Medicine for their creation. It is motivating to note that X-rays were discovered in 1895 by Wilhelm Conrad Roentgen, for which he established the 1901 Nobel Prize for Physics. These two inventions, nearly 100 years apart, led to some of the most active submission areas of image processing today [2].

From the 1960s until the present, the field of image processing has grown-up energetically. In addition to requests in medicine and the space package, digital image processing methods now are used in a broad range of applications. Computer events are used to improve the contrast or code the intensity levels into color for easier clarification of X-rays and other images used in production, medicine, and the biological disciplines. Geographers use the same or similar procedures to study pollution patterns from aerial and satellite imagery. Image improvement and rebuilding procedures are used to develop corrupted images of unrecoverable items or experimental results too luxurious to duplicate.

Today, there is almost no area of technical endeavour that is not impacted in some way by digital image processing. The areas of application of digital image processing are so varied that some form of organization is desirable in attempting to capture the breadth of this field. One of the simplest ways to develop a basic understanding of the extent of image processing applications is to categorize images according to their source (e.g., visual, X-ray, and so on). The principal energy source for images in use today is the electromagnetic energy spectrum. Other important sources of energy include acoustic, ultrasonic, and electronic (in the form of electron beams used in electron microscopy). Synthetic images, used for modelling and visualization, are generated by computer. Images based on radiation from the EM spectrum are the most familiar, especially images in the x-ray and visual bands of the spectrum.

The paper is organized as follows. In section II, we discuss related work with the motion estimation and compensation. In Section III, It describes introduction of video coding. In Section IV, it describes the introduction of motion estimation. Section V defines the proposed technique related to system. Finally, conclusion is given in Section VI.



II. RELATED WORK

S. Li et. al. [1] presented a pixel-based motion estimation scheme assisted with the coded texture video for depth inter-prediction, in view of motion similarity between depth and texture video. The proposed scheme can achieve higher inter-prediction gain without transmitting any motion vector in the pixel-based motion estimation. Coupled with depth-texture structure similarity, the inter prediction method is further extended to an integrated prediction approach by making use of both intra and inter information.

R. Li et. al. [2] proposed that the three-step search (TSS) algorithm has been widely used as the motion estimation technique in some low bit-rate video compression applications, owing to its simplicity and effectiveness. However, TSS uses a uniformly allocated checking point pattern in its first step, which becomes inefficient for the estimation of small motions. A new three-step search (NTSS) algorithm is proposed in this paper. The features of NTSS are that it employs a center-biased checking point pattern in the first step, which is derived by making the search adaptive to the motion vector distribution, and a halfway-stop technique to reduce the computation cost. Simulation results show that, as compared to TSS, NTSS is much more robust, produces smaller motion compensation errors, and has a very compatible computational complexity. L. Poet. al. [3] proposed a new four-step search (4SS) algorithm with center-biased checking point pattern for fast block motion estimation. Halfway-stop technique is employed in the new algorithm with searching steps of 2 to 4 and the total number of checking points is varied from 17 to 27. Simulation results show that the proposed 4SS performs better than the well-known three-step search and has similar performance to the new three-step search (N3SS) in terms of motion compensation errors. In addition, the 4SS also reduces the worst-case computational requirement from 33 to 27 search points and the average computational requirement from 21 to 19 search points as compared with N3SS. L. Liuet. al. [4] proposed a block-based gradient descent search (BBGDS) algorithm to perform block motion estimation in video coding. The BBGDS evaluates the values of a given objective function starting from a small centralized checking block. The minimum within the checking block is found, and the gradient descent direction where the minimum is expected to lie is used to determine the search direction and the position of new checking block. The BBGDS is compared with full search (FS), three-step search (TSS), one-at-a-time search (OTS), and new three-step search (NTSS). Experimental results show that the proposed technique provides competitive performance with reduced computational complexity. J. Thamet. al. [5] proposed that the widespread use of block-based inter-frame motion estimation for video sequence compression in both MPEG and H.263 standards is due to its effectiveness and simplicity of implementation. Nevertheless, the high computational complexity of the full-search algorithm has motivated a host of suboptimal but faster search strategies. A popular example is the three-step search (TSS) algorithm. However, its uniformly spaced search pattern is not well matched to most real-world video sequences in which the motion vector distribution is non-uniformly biased toward the zero vectors. Such an observation inspired the new three-step search (NTSS) which has a center-biased search pattern and supports a halfway-stop technique. It is faster on the average, and gives better motion estimation as compared to the well-known TSS. Later, the four-step search (4SS) algorithm was introduced to reduce the average case from 21 to 19 search points, while maintaining a performance similar to NTSS in terms of motion compensation errors. In this paper, they propose a novel unrestricted center-biased diamond search (UCBDS) algorithm which is more efficient, effective, and robust than the previous techniques. C. Zhuet. al. [6] proposed that in block motion estimation, a search pattern with a different shape or size has a very important impact on search speed and distortion performance. A square-shaped search pattern is commonly adopted in many popular fast algorithms. Recently, a diamond-shaped search pattern was introduced in fast block motion estimation, which has exhibited faster search speed. Based on an in-depth examination of the influence of search pattern on speed performance, we propose a novel algorithm using a hexagon-based search pattern to achieve further improvement. The hexagon-based search pattern is investigated in comparison with diamond search pattern, which demonstrates significant speedup gain over the diamond-based search. Analysis shows that a speed improvement rate of the hexagon-based search (HEXBS) algorithm over the diamond search (DS) algorithm can be as high as over 80% for locating some motion vectors in certain scenarios. In short, the proposed HEXBS algorithm can find a same motion vector with fewer search points than the DS algorithm. C. Cheung et. al. [7] proposed two cross-diamond-hexagonal search (CDHS) algorithms, which differ from each other by their sizes of hexagonal search patterns. These algorithms basically employ two cross-shaped search patterns consecutively in the very beginning steps and switch using diamond-shaped patterns. To further reduce the checking points, two pairs of hexagonal search patterns are proposed in conjunction with candidates found located at diamond corners. Experimental results show that the proposed CDHSs perform faster than the diamond search (DS) by about 144% and the cross-diamond search (CDS) by about 73%, whereas similar prediction quality is still maintained. M. Gallant et. al. [8] present an efficient computation constrained block-based motion vector estimation algorithm for low bit rate video coding that offers good tradeoffs between motion estimation distortion and number of computations. A reliable predictor determines the search origin. An efficient search pattern exploits structural constraints within the motion field. A flexible cost measure used to terminate the search allows simultaneous control of the motion estimation distortion and the computational cost. Experimental results demonstrate the viability of the proposed algorithm in low bit rate video coding applications,



achieving essentially the same levels of rate-distortion performance and subjective quality as that of the full search algorithm when used by the UBC H.263+ video coding reference software. However, the proposed motion estimation algorithm provides substantially higher encoding speed as well as graceful computational degradation capabilities.

I. Ahmad et. al. [9] proposed that Motion estimation (ME) is a multistep process that involves not one, but a combination of techniques, such as motion starting point, motion search patterns, and adaptive control to curb the search, avoidance of search stationary regions, etc. The collective efficiency of these techniques is what makes a ME algorithm robust and efficient across the board. This paper proposes a ME algorithm that is an embodiment of several effective ideas for finding the most accurate motion vectors (MVs) with the aim to maximize the encoding speed as well as the visual quality. The proposed algorithm takes advantage of the correlation between MVs in both spatial and temporal domains, controls to curb the search, avoids of search stationary regions, and uses switchable shape search patterns to accelerate motion search. The algorithm yields very similar quality compared to the full search but with several hundred times faster speed. They have evaluated the algorithm through a comprehensive performance study that shows that the proposed algorithm achieves substantial speedup without quality loss for a wide range of video sequences, compared with the ME techniques recommended by the MPEG-4 committee.

K. Ng et. al. [10] proposed that the fast motion estimation algorithms based on search point pattern are only good at handling videos with small motions, for example, block-based gradient descent search, diamond search and hexagonal-based search. An adaptive motion estimation algorithm which can switch between search patterns for different video contents should work better than a single search pattern algorithm. In this paper, a simple classifier based on Error Descent Rate (EDR) is proposed. This classifier uses a very few number of search points to predict whether the global minimum is far away or near the centre of the search window. If it is far away, a search pattern which is good at searching large motions is used. Otherwise, a pattern good at searching small motions is applied. The proposed search patterns switching (SPS) algorithm performs well for all kinds of video contents.

III. VIDEO PROCESSING

Digital video coding has regularly increased in importance since the 90s when MPEG-1 first emerged. It has had large influence on video delivery, storage and presentation. Compared to analog video, video coding achieves higher data compression rates without significant loss of subjective picture quality [3]. This eliminates the need of high bandwidth as required in analog video delivery. With this important characteristic, many application areas have occurred. For example, set-top box video playback using compact disk, video conferencing over IP networks, P2P video delivery, mobile TV broadcasting, etc. The specialized nature of video applications has led to the development of video processing systems having different size, quality, performance, power consumption and cost. In a video scene, data redundancy rises from spatial, temporal and statistical correlation between frames. These correlations are processed separately because of differences in their characteristics. Hybrid video coding architectures have been employed since the first generation of video coding standards, i.e. MPEG. MPEG consists of three main parts to decrease data redundancy from the three sources described above. Motion estimation and compensation are used to reduce temporal redundancy between successive frames in the time domain. Transform coding, also commonly used in image compression, is working to reduce spatial dependency within a frame in the spatial domain. Entropy coding is used to reduce statistical redundancy over the residue and compression data. This is a lossless compression technique commonly used in file compression. A major problem in a video is the high requirement for bandwidth. A typical system needs to send dozens of individual frames per second to create an illusion of a moving picture. For this reason, several standards for compression of the video have been developed. Each individual frame is coded so that redundancy is removed. Furthermore, between consecutive frames, a great deal of redundancy is removed with a motion compensation system.

A. Video Standards

Since there are endless ways to compress and encode data, and many terminal vendors which each may have a unique idea of data compression, common standards are required, that rigidly define how the video is coded in the transmission channel. There are mainly two standard series in common use, both having several versions. International Telecommunications Union (ITU) started developing Recommendation H.261 in 1984, and the effort was finished in 1990 when it was approved. The standard is aimed for video conferencing and video phone services over the integrated service digital network (ISDN) with bit rate a multiple of 64 kilobits per second. MPEG-1 is a video compression standard developed in joint operation by International Standards Organization (ISO) and International Electro-Technical Commission (IEC). The system development was started in 1988 and finished in 1990, and it was accepted as standard in 1992. MPEG-1 can be used at higher bit rates than H.261, at about 1.5 megabits per second, which is suitable for storing the compressed video stream on compact disks or for using with interactive multimedia systems [3]. The standard covers also audio associated with a video. In 1996 a revised version of the standard, Recommendation H.263, was finalized which adopts some new techniques for compression, such as half pixel and optionally smaller



block size for motion compensation. As a result it has better video quality than H.261. Recommendation H.261 divides each frame into 16×16 picture element (pixel) blocks for backward motion compensation, and H.263 can also take advantage of 8×8 pixel blocks. A new ITU standard in development is called H.26L, and it allows motion compensation with greater variation in block sizes.

For motion estimation, MPEG-1 uses the same block size as H.261, 16×16 pixels, but in addition to backward compensation, MPEG can also apply bidirectional motion compensation. A revised standard, MPEG-2, was approved in 1994. Its target is at higher bit rates than MPEG-1, from 2 to 30 megabits per second, where applications may be digital television or video services through a fast computer network. The latest ISO/IEC video coding standard is MPEG-4, which was approved in the beginning of 1999. It is targeted at very low bit rates (832 kilobits per second) suitable for e.g. mobile video phones. MPEG-4 can be also used with higher bit rates, up to 4 megabits per second.

IV. MOTION ESTIMATION

A video sequence can be considered to be a discretized three-dimensional projection of the real four-dimensional continuous space-time. The objects in the real world may move, rotate, or deform. The movements cannot be observed directly, but instead the light reflected from the object surfaces and projected onto an image. The light source can be moving, and the reflected light varies depending on the angle between a surface and a light source. There may be objects occluding the light rays and casting shadows. The objects may be transparent (so that several independent motions could be observed at the same location of an image) or there might be fog, rain or snow blurring the observed image. The discretization causes noise into the video sequence, from which the video encoder makes its motion estimations. There may also be noises in the image capture device (such as a video camera) or in the electrical transmission lines. A perfect motion model would take all the factors into account and find the motion that has the maximum likelihood from the observed video sequence. Changes between frames are mainly due to the movement of objects. Using a model of the motion of objects between frames, the encoder estimates the motion that occurred between the reference frame and the current frame. This process is called motion estimation (ME). Block-matching motion estimation (BMME) is the most widely used motion estimation method for video coding. Interest in this method was initiated by Jain and Jain and he proposed a block-matching algorithm (BMA) in 1981. The current frame is first divided into blocks of $M \times N$ pels. The algorithm then assumes that all pels within the block undergo the same translational movement. Thus, the same motion vector, d is assigned to all pels within the block. This motion vector is estimated by searching for the best match block in a larger search window pels centered at the same location in a reference frame [5].

A. Matching Criteria for Motion Estimation

Inter frame predictive coding is used to eliminate the large amount of temporal and spatial redundancy that exists in video sequences and helps in compressing them. In conventional predictive coding the alteration between the present frame and the projected frame is coded and conveyed. The better the prediction, the smaller the error and hence the transmission bit rate when there is motion in a sequence, then a pel on the same part of the moving object is a better prediction for the current pel [5].

1. Block Size

The important parameter of the BMA is the block size. If the block size is smaller, it achieves better prediction quality. This is due to a number of reasons. A smaller block size reduces the effect of the accuracy problem. In other words, with a smaller block size, there is less possibility that the block will contain different objects moving in different directions. In addition, a smaller block size provides a better piecewise translational approximation to non-translational motion. Since a smaller block size means that there are more blocks (and consequently more motion vectors) per frame, this improved prediction quality comes at the expense of a larger motion overhead. Most video coding standards use a block size of 16×16 as a compromise between prediction quality and motion overhead. A number of variable-block-size motion estimation methods have also been proposed in the literature. H.263 and MPEG standards allow adaptive switching between block sizes of 16×16 and 8×8 on a Macro Block (MB) basis.

2. Search Range

The maximum allowed motion displacement d_m , also known as the search range, has a direct impact on both the computational complexity and the prediction quality of the BMA. A small d_m results in poor compensation for fast-moving areas and consequently poor prediction quality. A large d_m , on the other hand, results in better prediction quality but leads to an increase in the computational complexity. A larger d_m can also result in longer motion vectors and consequently a slight increase in motion overhead. In general, a maximum allowed displacement of $d_m = \pm 15$ pels is sufficient for low-bit-rate applications. MPEG standard uses a maximum displacement of about ± 15 pels, although this range can optionally be doubled with the unrestricted motion vector mode [5].

3. Search Accuracy

Initially, the BMA was designed to estimate motion displacements with full-pel accuracy. Clearly, this limits the performance of the algorithm, since in reality the motion of objects is completely unrelated to the sampling grid. A

number of workers in the field have proposed to extend the BMA to sub-pel accuracy. For example, Ericsson demonstrated that a prediction gain of about 2 dB can be obtained by moving from full-pel to 1/8-pel accuracy.

V. PROPOSED WORK

In a typical Block Matching Algorithm, each frame is separated into blocks, each of which contains of luminance and chrominance chunks. Usually, for coding efficiency, motion approximation is achieved only on the luminance block. Each luminance block in the current frame is matched against applicant blocks in a search area on the reference frame. These applicant blocks are just the exiled versions of original block. The best candidate block is found and its movement (motion vector) is recorded. In a typical inter-frame coder, the input frame is deducted from the forecast of the reference frame. Therefore the motion vector and the resulting error can be conveyed instead of the original luminance block; thus inter-frame redundancy is removed and data firmness is attained. At receiver end, the decoder builds the frame difference signal from the received data and adds it to the rebuilt reference frames. The main objectives is to design an algorithm for detecting and extracting each frame in video sequence for detecting motion between these frames by improved background subtraction technique and to provide motion detection between extracted frames in real time moving objects using USB camera.

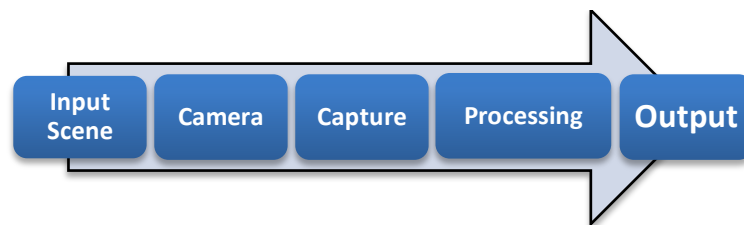


Figure 1: Steps in Coding

VI. CONCLUSION

Pixel motion estimation is a well-established approach. It provides a review on detecting and extracting each frame in video sequence for detecting motion between these frames by improved background subtraction technique. In this work, it proposed motion estimation while reconstruction is using block based approach so that our computation time gets decreased. It also proposes motion estimation with enhancement so that it adjusts contrast of input frames. It also proposes real time compressed video motion estimation. Motion Estimation forms a major computation bottleneck in video processing applications such as the detection of noise in image sequences, interpolation/ prediction of missing data in image sequences and de-interlacing of image sequences. The proposed algorithm can reduce the computational time as compared to block based technique.

REFERENCES

- [1] S. Li, J. Lei, C ZhuLuyu, and ChunpingHou, "Pixel-Based Inter Prediction In Coded Texture Assisted Depth Coding", IEEE Signal Processing Letters, Vol. 21, No. 1, January 2014.
- [2] R. Li, B. Zeng, and M. L. Liou, "A New Three-Step Search Algorithm For Block Motion Estimation," IEEE Transactions On Circuits And Systems For Video Technology, Vol. 4, No. 4, August 1994.
- [3] L.M. Po W.C. Ma, "A Novel Four-Step Search Algorithm For Fast Block Motion Estimation" IEEE Trans. Circuits Syst. Video Technol., Vol. 6, No. 3, Pp. 313-317, Jun. 1996.
- [4] L. Liu, E. Feig, "A Block-Based Gradient Descent Search Algorithm for Block Motion Estimation in Video Coding" IEEE Transactions On Circuits And Systems For Video Technology, Vol. 6, No. 4, August 1996.
- [5] J.Y. Tham, S. Ranganath, M. Ranganath, A. Kassim, "A Novel Unrestricted Center-Biased Diamond Search Algorithm For Block Motion Estimation", IEEE Transactions On Circuits And Systems For Video Technology, Vol. 8, No. 4, August 1998.
- [6] C. Zhu, X. Lin, L. Chau, "Hexagon-Based Search Pattern For Fast Block Motion Estimation", IEEE Transactions On Circuits And Systems For Video Technology, Vol. 12, No. 5, May 2002
- [7] C.H. Cheung, L.M. Po, "Novel Cross-Diamond-Hexagonal Search Algorithms for Fast Block Motion Estimation", IEEE Transactions On Multimedia, Vol. 7, No. 1, February 2005.
- [8] M. Gallant, F. Kossentini, "An Efficient Computation-Constrained Block-Based Motion Estimation Algorithm for Low Bit Rate Video Coding".
- [9] I. Ahmad, W. Zheng, J. Luo, M. Liou, "Fast Adaptive Motion Estimation Algorithm", IEEE Transactions On Circuits And Systems For Video Technology, Vol. 16, No. 3, March 2006.
- [10] K. Ng, L. Po, K.M. Wong, "Search Patterns Switching For Motion Estimation Using Rate of Error Descent", ICME 2007.
- [11] S. Goel, M. A. Bayoumi, "Multi-Path Search Algorithm for Block-Based Motion Estimation", ICIP 2006.
- [12] A. M. Tourapis, "Enhanced Predictive Zonal Search For Single And Multiple Frame Motion Estimation".
- [13] D. Kesrarat, V. Patanavijit, "Empirical Study on Performance Comparisons of Block-Based Motion Estimation on Multi Sub-Pixel Displacement with Multiples Block Size", 26th International Conference on Advanced Information Networking and Applications Workshops, 2012.
- [14] S. Klomp, M. Munderloh, Y. Vatis, J. Ostermann, "Decoder-Side Block Motion Estimation For H.264 / Mpeg-4 Avc Based Video Coding" IEEE Institute For Informations, 30167 Hannover, Germany 2009.