International Journal of Applied Engineering Research
ISSN 0973-4562 Volume 10, Number 11 (2015) pp. 29425-29438

http://www.ripublication.com

High Efficiency Video Coding Using NCDS Algorithm For Edge Preservation

D.Monisha[1] and G. Sethuram Rao[2]
[1]*Course of Embedded System Technologies, Velammal Institute of Technology, Panchetti-601204, TN, India*
[2]*Associate professor, Velammal Institute of Technology, Panchetti-601204, TN, India*
[1]*monicool74@gmail.com*, [2]*sethuramvit@gmail.com*

Abstract

In image processing, motion extraction (ME) plays a major role. It involves the compression of videos using macro blocks. A newer algorithm is used New Cross Diamond Search (NCDS) which is used for better edge preservation in ME. The operation of this algorithm is made suitable for both spatial and temporal domain. The working of the algorithm involves search points which is used for the detecting the images while sending large number of frames for better edge preservation. Due to its compact shape the power dissipation can be minimized and also used during the operation of video coding which helps in producing high efficiency video coding and also reduces its complexity. It generally involves 9 search points when compared to other algorithms. It also involves SAD for better computation.

Index Terms: feature extraction, motion estimation, new cross diamond search (NCDS), three step search (TSS), sum of absolute difference (SAD)

Introduction

In the past few decades, we have been running towards the Digital Age. Telephony networks, fiber optic networks, Internet, satellites, the third-generation (3G) wireless networks, these advanced transmission technologies and network infrastructure ensure digital information can flood into every corner around us. At the same time, the traditional media format has been gradually replaced by today's digital multimedia formats. These new formats and technologies shorten the distance between people, and improve the quality of lives.

In real time systems Motion Extraction (ME) plays a major role in which images are considered in the block size. In this system new cross diamond search algorithm is used to detect the images at the edges while sending in frames. With the help of these, edges are preserved. In the existing system, there includes three step search algorithm (TSS) which leads in excess time consuming, so it leads to complexity in developing. NCDS is mainly based upon search points. It also involves SAD (Successive Absolute difference) which leads in ease of computation. It leads in multiple search algorithms for detecting the images at the edges. It considers images in the form of macro block size. Motion extraction (ME) involves these types of techniques to make image efficient. It involves images in the form of 16x16 and is stored in the input buffer which is used during the SAD computation.

$$SAD = \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} |C_{ij} - R_{ij}|$$

where, C_{ij} is a block of current frame and R_{ij} is a block in the reference area.

In traditional methods there exists a full search (FS), diamond search (DS), cross diamond search (CDS), three step search (TSS), kite search (KS). These methods are efficient during the compression technique. Due to its reduced flexibility it cannot be used for all types of motion extraction (ME). It also considers many search points which make difficult during compensation technique. So in this proposed algorithm, it is designed to overcome all the needs which make video coding more efficient. It can also make ME more efficient in both spatial and temporal domain.

Related Work

In this section we can briefly discuss about the techniques which are involved in motion estimation

In the paper [10] we have discussed that, Image recognition is not only a part of video or motion application, it is also used in a wide range of multimedia application. It provides not only high performance at low cost but also provides high flexibility due to algorithmic diversity. The main tool or processor used is IMAP (integrated memory array processor) and also highly parallel SIMD processor is used. The efficiency of IMAP is evaluated by kernel benchmark and also for vision based application. In this image filtering is used which helps in reducing the noise level in an image. During the process of data intensity of image filtering the average speed was up to 3 and 8 times. The main drawback of using this IMAP architecture is that it uses the pure SIMD model which do not much at ease while using of processing elements (PE) array in image recognition. This ultimately leads to decreasing of efficiency. This inefficiency is mainly used in the vehicle detection application and this type of recognition is not suitable for other applications due to its inefficiency.

In the paper [1] we have discussed that Diamond Search (DS) algorithm is better than Four-step search (4SS) and Three Step Search (TSS) which is the main functionality carried in motion feature extraction. This is made sustainable when the performance result of Block-Based Gradient Descent Search (BBGDS) which reveals that, the performance of mean-square error performance is better when compared to other search step. It also reveals that the number of search points required for the feature extraction is also reduced which helps in time consumption.

From the observation of [9] it is inclined that, while compressing the videos of MPEG there finds an anomaly that TSS is mainly involved in low-bit rate of video compression. When TSS used for higher-bit rate does not provide high efficiency when compared to low-bit rate. So there exists New TSS (NTSS) which performs the operation similar to TSS which includes same search points, but with reduced cost when used for higher data compression.

From the [11] paper works, it is proved that, search points and motion vector distribution (MVD) has a greater impact in video coding techniques which ultimately results in the form of speed and quality of estimation. In this SCDS (Small CDS) is used which results in the execution of the MVD are in a faster manner and also includes minimal search points which satisfies the function of video coding.

Based on the experimental results of [13] it is known that, by varying the search points and check points which is used in ME from 2 to 4 and 17 to 27, it is summarized that 4SS perform better when compared to TSS and NTSS. It also provides a clear strategy that search points and check points are reduced from 33 to 27 and on an average, there exists only 21 to 19 rather than NTSS.

Strategy Selection

The search strategy following the prediction set is selected on the basis of the estimated motion activity measured as the standard deviation (Std. Dev.) of MVs of spatially neighboring MBs with respect to their median. Based on experiments, the Std. Dev threshold value is set for three to distinguish between high- and moderate/low-motion activity. For the sake of its wide range, three step search (TSS) [2] is selected to track high-motion activity MBs. For the rest of MBs, the diamond search (DS) [3] algorithm is selected. However, since the large diamond search pattern used by DS is rather sparse, kite-cross-diamond search (KCDS) [4] is employed when the number of SPs remaining after the evaluation of the prediction set (Number of SPs) for Integer-PEL ME (IPME) is smaller than 10 (actually 26 if Fractional-PEL ME is also taken into account). KCDS uses a denser search pattern than DS and, thus, is better suited to track small/moderate motion when the number of SPs is small. The whole MPS procedure with the search strategy selection is presented.

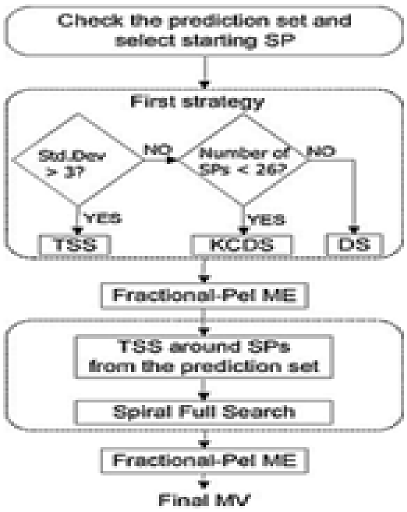

Figure 1: Flow Chart of MPS Procedure

Most of the fast ME algorithms are targeted to use the number of SPs as small as possible. It is a reasonable strategy when the computational resources are scarce. However, in a hardware video coding system, the number of available SPs can vary along with the resolution of currently processed video. For lower resolution and smaller search ranges, the number of available SPs per MB increases. If there is no necessity to save the energy, the extra computation can be used to improve the prediction efficiency, which is not possible when a non-scalable search strategy is employed. The proposed ME algorithm can terminate computations if the number of checked SPs is equal to the number assigned to a given macro block.

When the strategy following the prediction set is completed (e.g., TSS, DS, or KCDS) and SPs are still available, search paths starting from remaining points of the prediction set are checked. This approach is called the multipath search (MPS) in contrast to the more popular approach adopted in many fast ME algorithms that explore only one search path starting from the SP giving the smallest value of block difference measure. Since MVs from the prediction set are usually tightly clustered, a wide-ranging strategy is more appropriate than some narrow-range strategies such as DS. Thus, the search is performed using the TSS algorithm around each SP for the prediction set. When some SPs are still available after this stage, spiral full search starting from the middle of the search area can further improve the prediction efficiency (previously checked SPs are skipped).

A. Variable Block Size

Since optimal MVs of all partition modes are usually highly correlated, the probability of finding the optimal MV in the close neighborhood of MV for mode 16 × 16, is, on average, larger than 80% [7]. Therefore, similar to the fast full search adopted in the JM H.264/AVC reference software [15], each SP is checked for all modes in parallel. Particularly, all SADs of 4×4 blocks are computed and then reused to obtain SADs for other modes. However, after IPME, each node receives its own search center from which the process of Fractional- PEL ME (FPME) starts. This leads to a substantial increase in computational cost. If each of 41 partitions has a different search center for FPME and the hierarchical FPME is used (eight SPs for the half-PEL ME and eight for the quarter-PEL ME), 656 additional checks for FPME are needed. In the proposed approach, all modes check the same MVs and only SAD for mode 16 × 16 determines decisions within one common search path. As a consequence, search centers for Integer-PEL, Half-PEL, and Quarter-PEL ME for all modes are the same as that selected for mode 16 × 16. It makes it possible to check all modes in parallel also during FPME with relatively small coding-efficiency degradation

B. Fractional PME

Since FPME has a significant impact on prediction errors, the decision whether to continue IPME or switch to FPME has to be made. By using traditional algorithms such as TSS or DS, the choice is simple as FPME is introduced when the IPME procedure is completed. In the case of computationally scalable algorithms, the criterion is not so obvious since IPME can be continued up to FS. However, when resources are limited, it is better to switch to FPME earlier instead of continuing IPME with an in significant gain. Some experiments were performed to find such a moment. In particular, the switching was enforced at different search steps. It was found that FPME improves results if at least the first step of the first strategy (following the prediction set) is completed. However, if the number of still available SPs after the first step exceeds the minimum required for FPME (16 SPs), the IPME procedure is continued until the number falls to 16 or all steps of the first strategy are completed. If some SPs remain after the first FPME, IPME is resumed using spiral full search. However, each following IPME step is interrupted and FPME is invoked when the number of still available SPs falls to 16.

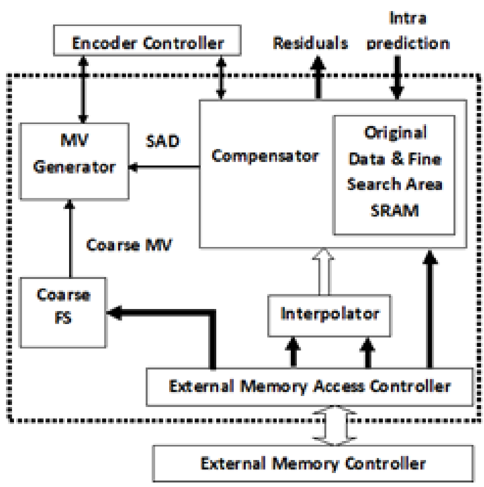

Figure 2: Motion-Estimation System Architecture

C. Multireference ME

Multireference ME can further improve the efficiency of inter-frame prediction. In H.264/AVC, up to 16 RPs can be searched in each direction. However, when an ME algorithm searches each RP separately, the computational complexity increases linearly with the number of RPs. In MPS with the Multireference extension, each RP is partially searched and then the decision is made as to which RP should undergo the detailed search. After the preliminary search, the RP that gives the smallest value of block difference measure undergoes the detailed check with further steps of the MPS algorithm without FS. If the amount of available resources is sufficient, other references can be searched as well. The order of search is determined based on the previously obtained values of block difference measure. The analysis presented in shows that prediction set and the first strategy have a major contribution to the final result. Thus, they are performed on the nearest RP with the quarter-pixel accuracy. This takes about 30 SPs per MB, on average. The MV found after this stage is included in the prediction set of the second RP. Since the nearest RP is selected as optimal in more than 50% of cases [16], it receives more SPs than others. In the second and remaining RPs, only SPs from the prediction set and the one step of the first strategy are checked with the quarter-pixel accuracy, which takes about 25 SPs, on average. Since the best match in further RPs often lies in the vicinity of the best MV found in the closest RP, it is sufficient to make a reliable decision which reference should be selected.

Proposed System

A. New Cross Diamond Search (NCDS) algorithm

In this section we first to distinguish the search pattern used in the algorithm and later the search path strategy will be excused.

Search Patterns

The hunt-point configuration used in the NCDS is divided in 2 different shapes: Cross-shaped pattern and diamond-shaped form. In this there exists some of the steps

Measure 1 (Starting - Small Cross Shape Pattern SCSP): A minimum BDM is used which is comprised of 5 search points which is sited at the heart of the search window.If the minimum BDM point occurs at the heart of the SCSP (0,0), the search stops, otherwise proceed to Step 2.

Measure 2 (SCSP): With the minimum vertex point at the nub, a new SCSP is formed. If there exists a minimum BDM point at the center the search is finished, otherwise proceed to Step 3.

Measure 3 (Guiding - Large Cross Shape Pattern - LCSP): The three unchecked outermost search points which are placed at the center of LCSP is checked, in which is used to guide the possible correct direction of the subsequent steps. And then proceed to Step 4.

Measure 4 (Diamond Searching): A new Large Diamond Search Pattern LDSP is formed by putting back the minimum BDM as the core of the LDSP.If the new

minimum BDM point occurs in the marrow, then go to Step 5 for the final solution; otherwise, this measure is repeated.
Measure 5 (Ending – Converging step): With the minimum BDM point in the previous step as the midpoint, a SDSP (Small Diamond-Shaped Pattern) is made.Place the new minimum BDM point from the SDSP, which is the final result of the motion vector.

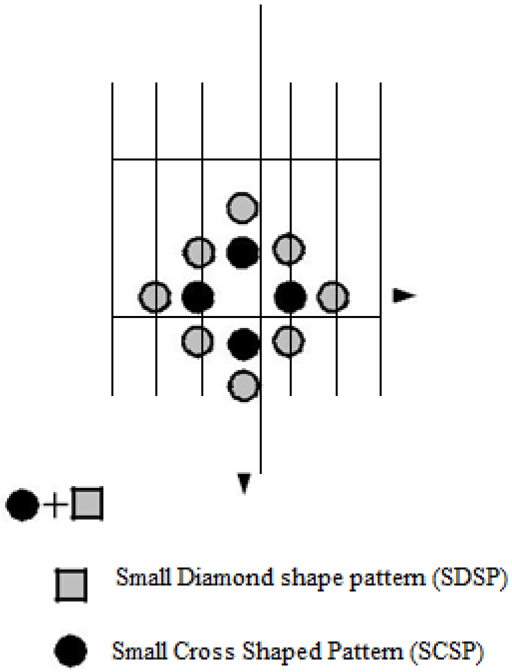

Figure 3: NCDS Pattern

Analysis of NCDS algorithm
To compare the CDS and the DS, the main improvement of this algorithm is the speed performance (the number of searching point). NCDS reduces the number of search points significantly if there is stationary block or quasi-stationary blocks. To accommodate the cross-center-based MV distribution characteristics, it offers more chance to spare up the searching points for motion vectors. In Fig 3, it shows 4 typical examples of NCDS and each candidate point is labeled with the corresponding step number. The NCDS only takes 5 (first step stop) and 8 (second step stop), whereas the CDS took 9 and 11 search points, and the DS took 13 search points for exploring the same block respectively. Referable to the hazard of being trapped to local minimum if the algorithm is keeping an SCSP in first and second search step, a guiding step in Step 3 of NCDS is trying to lead a possible correct direction by utilizing a larger search pattern when r > 1 (the minimum BDM point is still on the peak of the second SCSP in step 2). Thus, we employed LCSP step to avoid the algorithm being trapped by local minimum, which will influence the distortion performance. After step 3, the subsequent steps will be exactly the diamond search. This algorithm is suited for fast and a low bit-rate video conferencing application, which is easy and small movement.

B. Diamond search

Search patterns

In this algorithm the images are preserved more at the edges while TSS is used to prepare the video more efficient as it considers many search details. These search points are considered at the center point of the image. By this the images are considered at the mid centric of which the compression, efficient but leads to complex computation. In NCDS there includes large diamond search pattern (LDSP) and small diamond search pattern (SDSP).

The NCDS algorithm

Step 1: These diamond search makes more efficient as it considers only 9 search points where it four search points at corners and other four search points is at the midpoint of the images. It uses unrestricted center point of the image as indicated

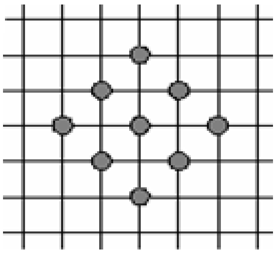

Figure 4: LDSP (Large Diamond Search Pattern)

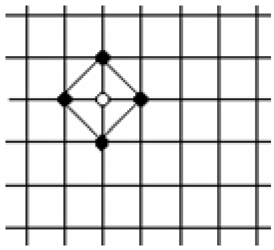

Figure 5: SDSP (Small Diamond Search Pattern)

Step 2 (BDS): when the new search points formed in LDSP seems to overlap, during evaluation only non-overlapping search points are used which is mainly used for reducing the search points. During LDSP the images are computed by TSS and are experimented until it satisfies Minimum Block Distortion (MBD) at the center point of the image. If MBD is achieved the loop closes

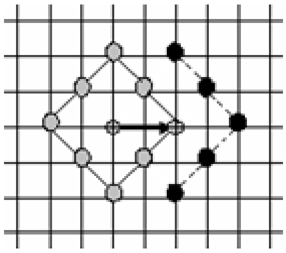

Figure 6: MBD point at corner

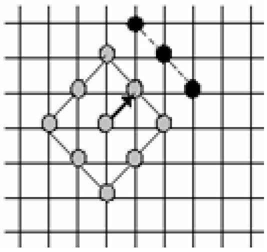

Figure 7: MBD points at the edges

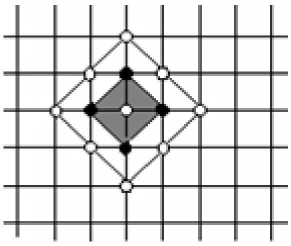

Figure 8: MBD at center

Step 3 (Ending): With the help of MBD motion vector is generated and by using many search points the vector is made more efficient and this leads to difficulty during computation. To overcome these disadvantage three step diamond search is used.

C. Three step search

The working of TSS algorithm is as same as the DS algorithm which varies only on step size. Based on MBD point the number of such points is set which varies in step size. In this the search point is reduced to three regardless of the location of MDB point. By reducing the step size it achieves SDSP pattern.

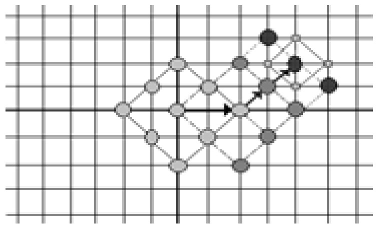

Figure 9: TSS Pattern

The LDSP pattern is repeated continuously until the MBD reaches the center of the image. It is mainly used for the compactness and the reduced search points. Some of the steps included are as follows

Step 1: Initially LDSP is created at the origin and viewed as a reference point. Now each point is searched with the help of LDSP. If the MBD is reached at the center, then follow step 3 else step 2 to be done to achieve MBD.

Step 2: Form a new LDSP viewing MBD as a search point and repeat step 1 if it is not achieved repeat the step once again.

Step 3: From the SDSP with the help of previous MBD as a center point. By this new MBD pattern is obtained which determines the motion vector (x, y). MBD technique can also be used as a search direction and also used as search point.

Results and Discussion

A. Performance comparison

A window size 15x15 is used for viewing the output with the help of MBD and keeping the LDSD as the origin point of searching. The performance is evaluated by error metrics namely error signal and signal to noise ratio.

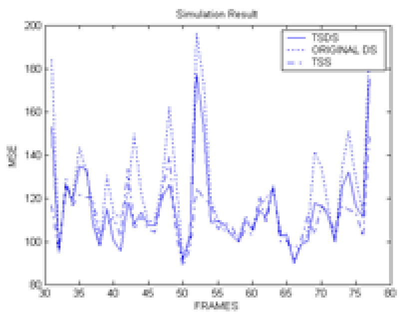

Figure 10: Mean Square Error

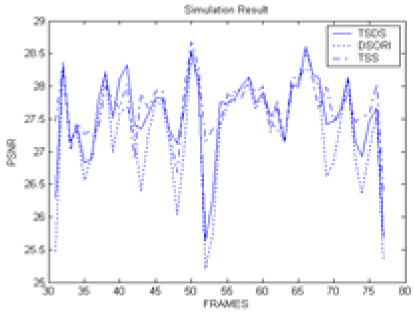

Figure 11: Signal To Noise Ratio

This result shows that it provides better performance when compared with traditional algorithms namely TSS and DS. It also provides simplicity and reduction in step size has an efficient implementation. By this motion vector is generated with the help of MBD.

B. Output parameters

This paper mainly discusses about the optimization of the algorithms which are used in feature extraction. This optimization mainly deals with the speed, area, power and delay. This approach can be dealt with the clear discussion.

I. Speed (fmax)

Timing analysis is a process of analyzing delays in a logic circuit to determine the conditions under which the circuit operates reliably. These conditions include, but are not limited to, the maximum clock frequency (fmax) for which the circuit will produce a correct output. Computing fmax is a basic function of a timing analyzer. The timing analyzer can be used to guide Computer-Aided Design tools in the implementation of logic circuits. For example, the circuit in Figure 1 shows an implementation of a 4-input function using 2-input AND gates. Without any timing requirements, the presented solution is acceptable. However, if a user requires the circuit to operate at a clock frequency of 250 MHz, the above solution is inadequate.By placing timing constraints on the maximum clock frequency, it is possible to direct the CAD tools to seek an implementation that meets those constraints.

II. Power dissipation

Device families have different power characteristics. There exist different parameters, namely power consumption, which includes process technology, supply voltage and electrical design and architecture. In device power consumption is the main function which varies differently based upon the device size. There exists two types of power consumption primary is static power and secondary is dynamic power. The device has major impact based on the dissipation of heat and also made a choice which has a major impact on cooling effect which improves a different variation on the power consumption. This variation shows a greater impact on static power due to its subthreshold leakage current and also varies with threshold voltage in a transistor. Process variation has a weak effect on dynamic power.

Fmax Summary

	Fmax	Restricted Fmax	Clock Name	Note
1	INF MHz	149.52 MHz	PE:PE1|res[0]	limit due to hold check
2	214.55 MHz	214.55 MHz	enable	

Figure 12: NCDS Fmax Summary

Fmax Summary

	Fmax	Restricted Fmax	Clock Name	Note
1	INF MHz	144.18 MHz	PE:PE1|res[0]	limit due to hold check
2	205.34 MHz	205.34 MHz	enable	

Figure 13: DS Fmax Summary

Fmax Summary

	Fmax	Restricted Fmax	Clock Name	Note
1	INF MHz	176.24 MHz	PE:PE1|res[0]	limit due to hold check
2	177.87 MHz	177.87 MHz	enable	
3	473.04 MHz	250.0 MHz	Clock	limit due to minimum period restriction (max I/O toggle rate)

Figure 14: TSS Fmax Summary

This table provides the clear optimization of the algorithms which is provided below

Table 1:

ALGORITHM	AREA	POWER	DELAY
NCDS	378	90.47 mw	214.55MHz
DS	383	139.40 mw	205.34MHz
TSS	392	205.34 mw	473.04MHZ

Comparison Chart

From this comparison chart we can conclude that NCDS algorithm has a better optimization when compared to the DS and TSS. So by using NCDS algorithm the motion extraction can be made more efficient manner.

Conclusion

Here we implemented a scalable Motion estimation Search algorithm for computationally efficient block motion estimation for image compression based motion speed. These techniques can be applied for both spatial and temporal image. Because of the compact shape of the search pattern and step size it outperforms other existing algorithms such as TSS, NCDS, and DS in terms of computational efficiency with a better performance. This algorithm can be used in video coding standards such as MPEG-4, H.264 AVC because of its ease of implementation, better performance with reduced computational complexity and high speed searching. To improve the visual quality and smoothing of edges between the macro blocks a de-blocking filter is used in the decoded video. This filter aims to improve the appearance of decoded pictures. Decoded frames consist of corner artifacts that reduce the quality of an image. To avoid these artifacts, many de-blocking filters are proposed, butit's failed to remove ringing artifacts, also the pixels are blurred due to non-uniform filter coefficients equalization. To overcome this issue we design an Adaptive bilateral loop filter that can apply across both vertical and horizontal position of the pixel blocks.

References

[1] P. G. Howard and J. S. Vitter, “Fast and efficient lossless image compression,” in Proc. IEEE Int. Conf. Data Compression, 1993, pp. 501–510.

[2] J. R. Jain and A. K. Jain, “Displacement Measurement and its application in Interframe Image Coding,” IEEE Trans, Communication, Vol.COM- 2 9, pp. 1799-1808, Dec 1981.

[3] T. Koga, K. Iinuma, A. Hirano, Y. Iijima, and T. Ishiguro, “Motion Compensated Interframe Coding for Video Conferencing,” Proc. Of NTC’81, New Orleans, LA, pp. G5.3.1-G5.3.5, Nov. 2 9-Dec. 3, 1981

[4] Shan Zhu and Kai-Kuang Ma, “A New Diamond Search Algorithm for Block Matching Motion Estimation,” IEEE Transactions on Image Processing, vol.9.pp. 2 87- 2 90, Feb 2000.

[5] Tsung-Han Tsai, Yu-Hsuan Lee, and Yu-Yu Lee, “Design and Analysis of High-Throughput Lossless Image Compression Engine Using VLSI-Oriented FELICS Algorithm,” IEEE Trans. Very Large Scale Integr. (VLSI) Syst., Vol. 18, No. 1, January 2 010.

[6] S. Mitra and T. Acharya, “Gesture Recognition: A Survey,” IEEE Trans. Syst., Man, Cybern. C, Appl. Rev., vol. 37, no. 3, pp. 311-324, May 2007.

[7] X. Wu and N. D. Memon, “Context-based, adaptive, lossless image coding,” IEEE Trans. Commune., vol. 45, no. 4, pp. 437-444, Apr. 1997.

[8] M. J. Weinberger, G. Seroussi, and G. Sapiro, “The LOCO-I lossless image compression algorithm: Principles and standardization into JPEG-LS,” IEEE Trans. Image Process., vol. 9, no. 8, pp. 1309–13 24, Aug. 2000.

[9] R. Li. B. Zeng, and M. L. Liou, "A New three-step search algorithm for block motion estimation", IEEE Transactions on circuits and systems on video technology, vol.6.pp.438-44 2, August 1994.

[10] S. Kyo, S. Okazaki and T. Arai, "An Integrated Memory Array Processorfor Embedded Image Recognition Systems," IEEE Trans. Comput., vol. 56, no. 5, pp. 622-634, May 2007.

[11] C. H. Cheung and L. M. PO, "A novel small-cross-diamond searchalgorithm for fast video coding and video conferencing applications," in Proc. IEEE ICIP, Sept. 2002.

[12] J. Y. Tham, S. Ranganath, M. Ranganath, and A. A. Kassim, "Anovel unrestricted center-biased diamond search algorithm forblock motion estimation," IEEE Trans, Circuits Syst. VideoTechnol., vol. 8, no. 4, pp. 369-377, Aug 1998.

[13] L. M. Po and W. C. Ma, "A novel four-step search algorithm for fast block motion estimation," IEEE Trans. Circuits Syst. Video Technol., vol. 6, pp. 313-317, Jun 1996

[14] C. H. Cheung, and L. M. Po, "A Novel Cross-Diamond SearchAlgorithm for Fast Block Motion Estimation," IEEE Trans,Circuits Syst. Video Technol., vol. 12, no. 12, Dec 2002.

[15] Yao Nie; Kai-Kuang Ma, "Adaptive rood pattern search for fastblock-matching motion estimation," IEEE trans. Image Processing,vol. 11, pp. 1442-1449, Dec 2002