NEW IMPROVED 2D SVD BASED ALGORITHM FOR VIDEO CODING

Sachin V[1] , Rahul Vasist, Shashank Hegde, Shashikiran and M Uttarakumari[2]

[1]Student, Department of Electronics and Communication Engineering, RVCE, Bangalore
sachinshetty001@gmail.com
[2]Department of Electronics and Communication Engineering, RVCE, Bangalore
uttarakumari@rvce.edu.in

ABSTRACT

Video compression is one of the most important blocks of an image acquisition system. Compression of video results in reduction of transmission bandwidth. In real time video compression the incoming video data is directly compressed without being stored first. Therefore real time video compression system operates under stringent timing constraints. Current video compression standards like MPEG, H.26x series, involve emotion estimation and compensation blocks which are highly computationally expensive and hence they are not suitable for real time applications on resource scarce systems. Current applications like video calling, video conferencing require low complexity video compression algorithms so that they can be implemented in environments that have scarce computational resources (like mobile phones). A low complexity video compression algorithm based on 2D SVD exists. In this paper, a modification to that algorithm which provides higher PSNR at the same bit rate is presented.

KEYWORDS

2D SVD, low complexity video compression

1. INTRODUCTION

Video is a sequence of still images representing scenes in motion. In a video, the images called as frames are streamed at the rate of 25-30 frames per second (fps). Video is characterized by huge amount of data. An uncompressed CIF video at a resolution of 288 X 352 at 25 fps has a data rate of30.41Mbits/s. As a result, transmission of raw video requires huge bandwidth. Also, the memory required to store this uncompressed video is enormous. These two drawbacks make it impractical to use raw video. To reduce the transmission bandwidth and the storage requirements, video compression is done. During compression, the redundant information is removed. Compression algorithms typically exploit spatial, temporal and psycho-visual redundancies present in a video. Some of the widely used video compression algorithms are MJPEG, MPEG-series and H.26x series. In MJPEG (MotionJPEG), each frame is individually coded using the JPEG algorithm. MJPEG does not exploit the temporal redundancies in the video and therefore it results in lower compression. A compression ratio of around 10:1 to 15:1 can be achieved using MJPEG without introducing any visual artifacts. MPEG (Moving Pictures Experts Group) is an experts group set by ISO and IEC. They have come up with standards like MPEG-1, MPEG-2, MPEG-4 etc. which have been widely used for video compression. The heart of MPEG or H.26x algorithm is the motion estimation and motion compensation block. Motion estimation and compensation is responsible for exploiting the temporal redundancies in the video. Here, rather than coding each block independently, a block in the current frame is used to find the same block

Natarajan Meghanathan, et al. (Eds): SIPM, FCST, ITCA, WSE, ACSIT, CS & IT 06, pp. 461–469, 2012.

DOI : 10.5121/csit.2012.2345

in the previous frame. Rather than sending the entire block, only the error and the motion vectors are encoded and sent to the decoder. Therefore the block can be represented with lower number of bits. Motion estimation and compensation is an efficient algorithm and a compression anywhere between 30:1 to 100:1 can be achieved. However, it is highly computationally complex. Due to the high computationally complexity, the power consumption is increased thereby reducing the battery life. Also, hardware implementation becomes difficult. Lot of research is being carried out in this field to reduce the computational complexity of the aforementioned algorithm. Chanyul Kim et al. [1] have designed a moving edge detection algorithm using DCT coefficients. Using this algorithm, motion estimation can be applied can be applied only to those blocks that contain a moving edge. This algorithm has been reported to take only5% of the original encoding time. K. Virk et al. [2] in their paper have proposed an improvement to 3DRS block matching algorithm. They have reported 50% reduction in the number of computations. Some of the other algorithms which have been known to reduce the number of computations are Three Step Search (TSS), Four Step Search (4SS), Diamond Search (DS), and Adaptive 1 Rood Pattern Search (ARPS).

Although many algorithms which significantly reduce the number of computations have been reported in literature, motion estimation is still considered a resource and power hungry part of the compression algorithm. Therefore, algorithms which employ motion estimation cannot be used in low power, resource scarce systems. Today, video calling and video conferencing using smart phones is becoming ubiquitous. Therefore there is a need for algorithms, which achieve the necessary compression in real time with low power consumption. This has led to the development of many algorithms which exploit the temporal redundancies of the video without resorting to motion estimation. Tarek Ouni et al. [4] have developed a novel algorithm in which the temporal redundancy is converted into spatial redundancy by accordion representation. JPEG algorithm is applied on this accordion representation to obtain higher compression ratio. Raymond Westwater et al. [5] have developed XYZ algorithm based on 3D DCT. Here, the normal 2D DCT is extended to the time axis to obtain 3D DCT. This results in compression which is comparable to that obtained using MPEG algorithms, however the number of computations is significantly less. Anew paradigm called Distributed Video Coding (DVC) was developed around 2002. These were based on two theorems .i.e. Slepian-Wolf and Wyner-Ziv theorem. In DVC, the complexity is shifted from the encoder to the decoder by the usage of error correcting codes (ECC). The state-of-the-art low complexity encoder is considered to be the DISCOVER Codec, which is based on the DVC paradigm. However DVC algorithms require a feedback channel from the decoder to the encoder. In this paper, an improvement to the 2D SVD algorithm proposed by Zhouye et al. [3] has been presented.

2. 2D SINGULAR VALUE DECOMPOSITION

In Linear Algebra, Singular Value Decomposition is nothing but factorization of a matrix in the form [6]

$$\mathrm{A} = U\Sigma V^T$$

where U and V are orthogonal matrices and Σ is a diagonal matrix .i.e.

$$A = (orthogonal)(diagonal)(orthogonal)$$

The columns of U are composed of the eigenvectors of A^T , the columns of V are composed of the eigenvectors of A^TA. The diagonal values of Σ are nothing but the square roots of the non-zero eigenvalues of both $AA^T$and A^TA.2D-SVD is an extension of the above mentioned 1DSVD.2D-SVD has been extensively studied for computer vision. The main drawback of 1D-SVD for image compression applications is that, even though it provides the most energy compaction (the coefficients are present only along the diagonal), it requires the transmission of the two eigenvector matrices for each block. This incurs very high overhead thereby reducing the compression efficiency. In 2D-SVD, the eigenvector matrices are extracted from a group of blocks. Therefore, the two matrices have to be transmitted only for a group of blocks. This results in higher compression.

2.1. 2D SVD Process

The basic algorithm to find the coefficient matrix is given below [8]:

1) Given a group of frames $A_1 \dots\dots A_n$, find the mean frame A$_{mean}$. Obtain the mean subtracted frames .i.e. $A_i' = A^i - A_{mean}$
2) For any GOB (Group of Blocks), B in the mean-subtracted frames, we denote as each block in B as $b_1 \dots\dots b_n$. For each GOB, find the row-row and column-column covariance matrices F and G

$$F = \sum_{i=1}^{n} b_i b_i^T$$

$$G = \sum_{i=1}^{n} b_i^T b_i$$

   U$_l$ and U$_r$ are made up of the k principal eigenvectors of F and s principal eigenvectors of G respectively. It has been reported in [3], [9] that the lowest mean squared error is obtained when k=s=1. Therefore the same is used in this project
3) The coefficient matrix M is obtained using the formula

$$M_i = U_l^T b_i U_r$$

   it is to be noted that M$_i$ is not a diagonal matrix. However, most of the non-zero coefficients will be located close to the principal diagonal
4) To get back the original frame, the mean subtracted block is first obtained using the formula

$$b_i = U_l M_i U_r^T$$

   The near optimal approximation of each block is obtained using

$$b_i'' = b_i' + b_{mean}$$

   where b$_{mean}$ is the corresponding block in the mean frame

3. PROPOSED ALGORITHM

The application of the algorithm proposed in [3] yielded the following observation.

1) For videos with low motion, increase in GOP increased the compression ratio while maintaining the PSNR almost constant. This is because the percentage of GI to the coefficients is reduced, while maintaining the energy compaction property of SVD.
2) For videos with high motion, increase in GOP decreased the compression ratio and the PSNR also reduced. This is because the energy is distributed over a large number of coefficients thereby reducing the coding efficiency.

From the above observation it is clear that, for videos with low motion, a large GOP can be used and for videos with high motion a very low value of GOP must be used to maintain the compression ratio and the PSNR. The proposed algorithm is based on this inference:

1) First, the incoming images are divided into 8X8 non-overlapping blocks, B$_j$ where j=0…$\frac{height*width}{64}$
2) For each block B$_j$, the difference between the current block and the corresponding block in the 10th next frame is computed. The sum of these differences is computed.

3) If this sum is less than the threshold *th*, then the GOP value is set to GOP$_{high}$. If it is greater than the threshold, then it is set to GOP$_{low}$. For each block, a group related information is first sent to the decoder. In this GI, the first bit set to 1 if GOP = GOP$_{high}$ and it is set to 0 if GOP = GOP$_{low}$
4) The mean block block b$_{mean}$, is calculated using the formula

$$b_{mean} = \frac{1}{GOP}\sum_{i=1}^{GOP} b_i$$

The mean block is then encoded using the JPEG algorithm. It is to be noted that, GOP$_{high}$ is chosen to be an integral multiple of GOP$_{low}$. If the GOP is GOP$_{low}$ then GOP$_{high}$ GOP$_{low}$ number of mean blocks are sent.

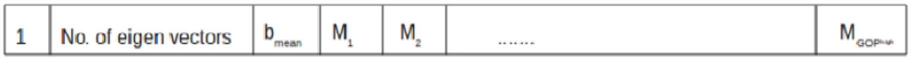


Figure 1: An encoded packet for GOP = GOP$_{high}$

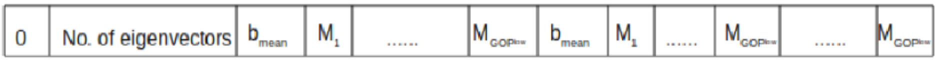


Figure 2: An encoded packet for GOP=GOP$_{low}$

5) 2-D SVD mentioned above is used to obtain the corresponding eigenvector matrices U_l^j and U_r^j and the groupof 8 X 8 coefficient matrices $M_1^j \ldots M_n^j$
6) The eigenvectors $U_l^j = [u_1^j \ldots u_8^j]$ and $U_r^j = [v_1^j \ldots v_8^j]$, are encoded using the Vector Quantization strategy. Codebooks of length 256, 256, 128, 128, 128, 64, 64, and 32 are used to quantize the eigenvectors respectively. Eigenvectors derived by applying the 2D SVD algorithm to some standard sequences are used to learn the codebooks. The LBG algorithm is used to learn these codebooks. These codebooks are stored in both encoder and decoder, therefore the coding of the eigenvectors is achieved using the least amount of bits possible.
7) There is no need to transmit all the obtained eigenvectors. Let $m_i^j\ (x,y)$ denote the coefficient value at frame i of block j at position (x,y). If X$_{max}$ denotes the maximum x position of the non-zero coefficients for block j and frames i...N and Y$_{max}$ denotes the maximum y position of the non-zero coefficients, then we need to transmit only the eigenvectors $[u_1^j \ldots u_{Xmax}^j]$ and $[v_1^j \ldots v_{Ymax}^j]$. Six bits are included in GI, to denote the number of eigenvectors sent per GOB.
8) The coefficient matrices $M_1^j \ldots M_n^j$ are then quantized, zigzag read and entropy encoded using the JPEG algorithm

An encoded packet for GOP = GOP$_{high}$ is shown in Figure 1 and the encoded packet for GOP=GOP$_{low}$ is shown in Figure 2. Figure 3 shows the 10th frame of Akiyo sequence, in which all the blocks for which GOP was set as GOP$_{high}$ has been displayed in black. The threshold was set at 25.

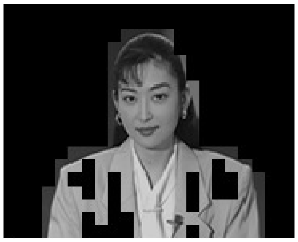


Figure 3: 10th frame of Akiyo. The blocks that are displayed in black are those for which the GOP was set as GOP_{high}

4. RESULTS

The 2D SVD algorithm given in [3] and our proposed algorithm were applied to the qcif test sequences Akiyo, Claire, Grandma. The video was considered to be grayscale and the algorithm was applied only to the Y part of the test sequences. However, the results obtained are applicable to U and V part also. The results shown were obtained by using the Quantization matrix shown in Figure 4.

$$Q_{\text{Table}}=\begin{bmatrix} 16 & 11 & 10 & 16 & 24 & 40 & 51 & 61 \\ 12 & 12 & 14 & 19 & 26 & 58 & 60 & 55 \\ 14 & 13 & 16 & 24 & 40 & 57 & 69 & 56 \\ 14 & 17 & 22 & 29 & 51 & 87 & 80 & 62 \\ 18 & 22 & 37 & 56 & 68 & 109 & 103 & 77 \\ 24 & 35 & 55 & 64 & 81 & 104 & 113 & 92 \\ 49 & 64 & 78 & 87 & 103 & 121 & 120 & 101 \\ 72 & 92 & 95 & 98 & 112 & 100 & 103 & 99 \end{bmatrix}$$

Figure 4: Quantization Matrix

For the 2D SVD algorithm, the GOP was set as 16, as it was found that this provided the highest PSNR for a given bitrate for the specified sequences (Akiyo, Claire, and Grandma). Figure 5 shows the plot of PSNR vs. Bitrate for varying GOP, when the proposed algorithm is applied to the first 192 frames of Claire. From this plot it is clear that the highest PSNR is obtained when GOPhigh = 48 and GOPlow = 4

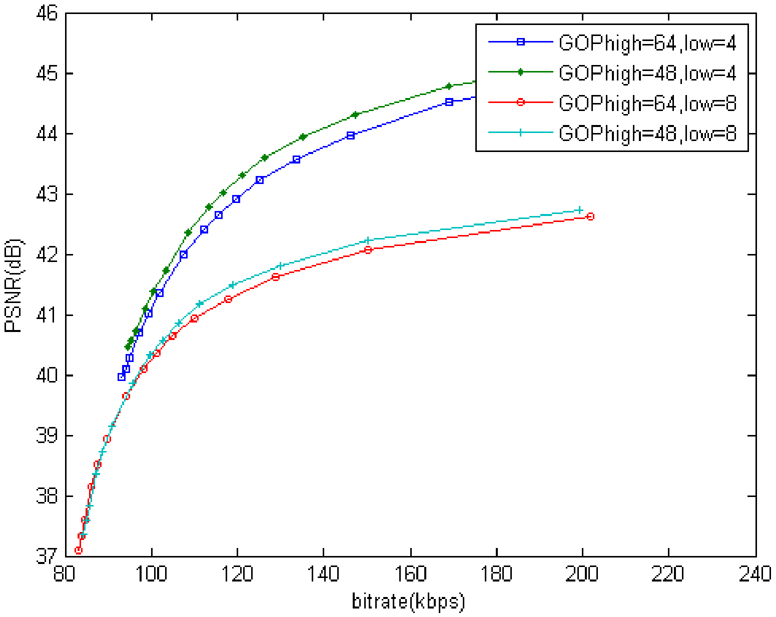


Figure 5: PSNR vs. Bitrate for varying GOP (first 192 frames of Claire)

Fig 6 gives the plot of PSNR vs. Bitrate for the first 192frames of the Akiyo test sequence. Fig 7 gives the same plot for Claire test sequence and Fig 8 for Grandma Test sequence. To obtain these results, the GOP of 2D SVD algorithm was set as 16 and for the proposed algorithm, GOP_{high} was chosen as 48 and GOP_{low} was chosen as 4. It can be seen clearly from these figures that, our proposed algorithm gives almost 2dB higher PSNR at the same bit rate.

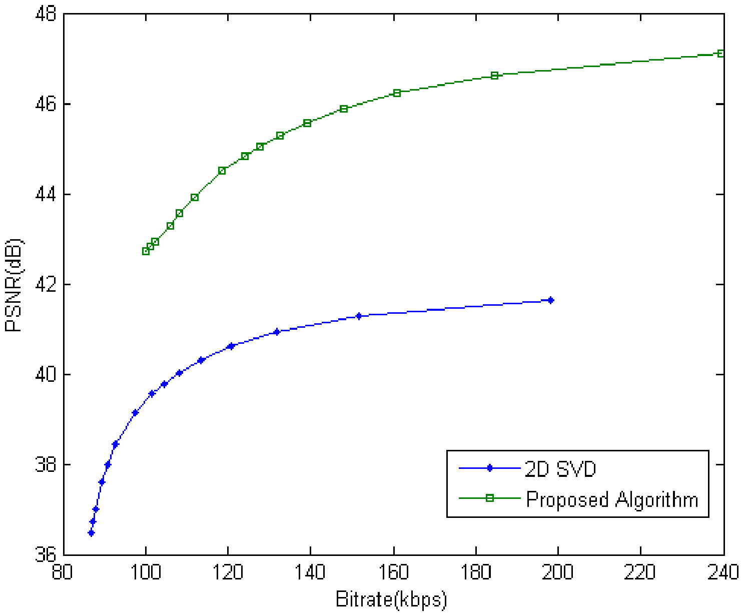


Figure 6: PSNR vs. Bitrate for Akiyo

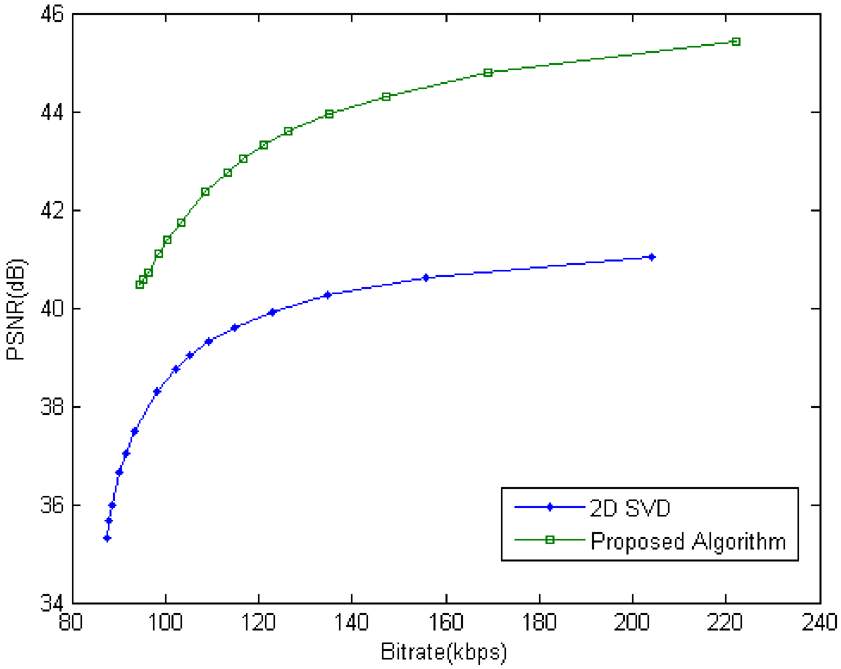


Figure 7: PSNR vs. Bitrate for Claire

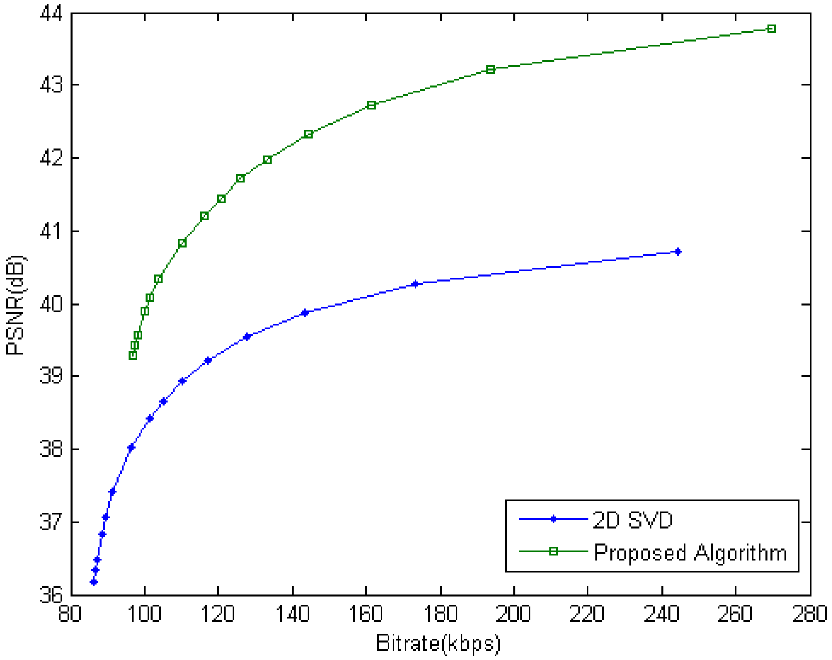


Figure 8: PSNR vs. Bitrate for Grandma

4. CONCLUSION

A new algorithm based on the 2D SVD algorithm proposed in [3] was presented in this paper. The proposed algorithm outperformed the existing algorithm by almost 2dB. This algorithm is highly suitable for video sequences in which motion is restricted to only a part of the frame. However, the performance improvement for sequences that has considerable movement is negligible. In the presented algorithm, GOP was given only 2 values. The algorithm can further be improved by using a range of values for GOP based on the amount of motion present in the block.

REFERENCES

[1] Chanyul Kim and Noel E. O' Connor, "Low Complexity Video Compression Using Moving Edge Detection Based on DCT Coefficients", 15th International Multimedia Modelling Conference (MMM 09), Jan. 2009

[2] K. Virk, N. Khan, S. Masud, F. Nasim, S. Idris, "Low Complexity Recursive Search Based Motion Estimation Algorithm for Video Coding Applications", Proc. of the 13th European Signal Processing Conference (EUSIPCO 2005), Sept. 2005

[3] Zhouye Gu, Weisi Lin,Bu-sung Lee, ChiewTong Lau, "Low-Complexity Video Coding Based on Two-Dimensional Singular Value Decomposition", IEEE Transactions on Image Processing, vol. 21, no. 2, February 2012, pp 674-687

[4] Tarek Ouni, Walid Ayedi, Mohamed Abid, "New low complexity DCT based video compression method", International Conference on Telecommunications, ICT(09), May. 2009, pp 202-207

[5] Raymond Westwater, Borko Furht "Real Time Video Compression techniques and algorithms", Kluwer Academic Publishers, 1997

[6] Gilbert Strang "Linear Algebra and its Applications", Thomson Brooks/Cole

[7] Rafael C. Gonzalez, Richard E. Woods "Digital Image Processing", Pearson Education, third edition, 2009

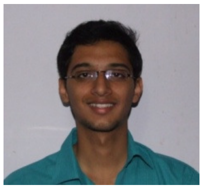
[8] C. Ding and J. Ye, "Two-dimensional singular value decomposition (2DSVD) for 2-D maps and images," in Proc. SIAM Int. Conf. Data Mining, 2005, pp. 32–43.

[9] J. Ye, "Generalized low rank approximations of matrices," in Proc. Int. Conf. Mach. Learn., 2004, pp. 887–894.

Authors

Mr **Sachin V** obtained his Bachelor of Engineering degree in Electronics and communication department from R.V. college of Engineering, Bangalore, India, in 2012. He co-authored a paper on ADSL presented at national conference NCETIT-2012 held at Bangalore. His area of interest includes Image processing, Neural networks, Digital communication. His immediate plan is to begin to work as Processor application engineer with Analog Devices Inc, Bangalore, India.

Mr **Rahul R Vasist** obtained his Bachelor of Engineering degree in Electronics and Communication department from R. V. College of Engineering, Bangalore, India in 2012. He is currently pursuing Master of Science in Embedded Systems in University of Pennsylvania, Philadelphia, USA.
His area of interest includes Embedded Systems, Image and Signal Processing.



Mr **Shashank S Hegde** obtained his Bachelor of Engineering degree in Electronics and communication department from R.V. college of Engineering, Bangalore, India, in 2012. He was a member of satellite project “studsat” conducted by ISRO. His area of interest includes Image processing, Neural networks, Embedded systems. He is currently working in IBM India Pvt. Ltd.

Mr **Shashikiran R Pargaonkar** obtained his Bachelor of Engineering degree in Electronics and Communication Engineering from R.V. College of Engineering, Bangalore, India, in 2012. He has participated in various IEEE Parliament debates. His area of interest includes Digital Electronics, Control systems and Neural Networks. He is a member of IETE.

Dr. **M.Uttara Kumari**, Professor & Head of the Department of Electronics & Communication at RV College of Engineering, Bangalore, India. She has obtained her doctoral degree in the 2007 from Andhra University, Vishakhapatnam, Masters degree in Electronics from Bangalore University in 1996 and BE in Electronics & Communication Engineering from Nagarjuna University Guntur, in the year 1989. Her research interests are Signal Processing, Communication. She is a member of IEEE, ISTE and IETE. She has published papers at various journals and International conference.