International Journal of Electronics and
Communication Engineering (IJECE)
ISSN(P): 2278-9901; ISSN(E): 2278-991X
Vol. 3, Issue 4, July 2014, 177-186
© IASET

International Academy of Science, Engineering and Technology
Connecting Researchers; Nurturing Innovations

OVERVIEW AND IMPLEMENTATION OF INTRAPREDICTIONS FOR H.264/AVC VIDEO CODEC

IMRAN ULLAH KHAN[1] & M. A. ANSARI[2]

[1]Research Scholar, Department of Electronics & Communication Engineering, Mewar University, Chittorgarh, Rajasthan, India

[2]Senior Member IEEE, Islamic University, Madinah, Medina, Saudi Arabia

ABSTRACT

The H.264 is an international video coding standard developed by the ITU-T ISO/IEC Joint Video Team (JVT).Video compression algorithms operate by removing temporal and spatial redundancies in a video encoder conforming to standards such as H.264. Intra Prediction modes in H.264 improve the compression, exploiting spatial redundancy. This paper presents the algorithm for all the modes of *intraprediction.* The simulated results show better compression can be achieved in Vertical, Horizontal and Horizontal up modes than other modes for a number of pictures that have been experimented with. The compression achieved was 10 or more for these modes without sacrificing the quality of the reconstructed picture. Quality achieved was over 32 dB, indistinguishable from the original. In most of the pictures experimented with, compression improves appreciably with *Intraprediction* than without it up to 51%.

KEYWORDS: *Intraprediction, Quantization, Cavlc, Macroblock Video Coding Layer (VCL)*

INTRODUCTION

Additional Features of H.264/AVC video Codec which distinguish it from the previous video compression standards are listed below [3, 4]:

- Two context adaptive coding schemes, CAVLC (context-adaptive variable-length coding) and CABAC (context-adaptive binary arithmetic coding), improve coding efficiency by adjusting the code tables according to the surrounding information.
- B-frame can be used as reference frame.
- Multiple reference frame motion compensation is allowed, which also improves the prediction accuracy. The restriction that only the immediate previous frame can be used as reference frame is thus removed.
- More motion compensation block sizes and shapes, such as 8x4, 4x8 and 4x4, are supported. The minimum luma motion compensation block size can be as small as 4x4.
- ¼ pixel motion estimation improves prediction accuracy. It has the same prediction accuracy as in MPEG-4 [6], but with lower interpolation complexity.
- Directional spatial prediction is applied in intra-coded macroblocks (MBs) of pictures to reduce the amount of information before their block transform.

- In-loop deblocking filtering removes the blocking artifacts caused by transform and quantization.
- Small block-size transform of 4x4 is used rather than 8x8 used in earlier standards, which results in less ringing artifacts.
- The Discrete Cosine Transform (DCT) is replaced by integer transform that is exact- match inverse transform, thus avoiding drift during inverse transform.
- Parameter sets are used between the encoder and decoder to achieve synchronization in terms of syntax.
- Flexible macroblock ordering (FMO) partitions a frame into different slice groups.
- Data partitioning groups a slice in up to three packets by their importance.
- SP/SI synchronization/switching frames reduce the penalty of switching between ongoing video bitstreams by avoiding transmission of an I-frame.
- Encoder can send redundant representation of some regions of a frame to enhance robustness to data loss.

OVERVIEW OF H.264 VIDEO CODEC

Comparing the H.264/AVC video coding tools like multiple reference frames, quarter-pixel motion compensation, deblocking filter or integer transform to the tools of previous video coding standards, H.264/AVC achieved a leap in coding performance. For efficient transmission in different environments not only coding efficiency is relevant, but also the seamless and easy integration of the coded video into all current and future protocol and network architectures. This includes the public Internet, as well as wireless networks expected to be a major application for the new video coding standard. The adaptation of the coded video representation or bit-stream to different transport networks was typically defined in the systems specification in previous MPEG standards or separate standards like H.320 or H.324. However, only the close integration of network adaptation and video coding can bring the best possible performance of a video communication system. Therefore H.264/AVC consists of two conceptual layers. The video coding layer (VCL) defines the efficient representation of the video, and the Network Abstraction Layer (NAL) converts the VCL representation into a format suitable for specific transport layers or storage media. For circuit-switched transport like H.320, H.324M or MPEG-2, the NAL delivers the coded video as an ordered stream of bytes containing start codes such that these transport layers and the decoder can robustly and simply identify the structure of the bit stream. For packet switched networks like Real-time Transport Protocol (RTP/IP) or Transmission Control Protocol/Internet Protocol (TCP/IP), the NAL delivers the coded video in packets without these start codes [5]. Figure 1 shows a basic H.264 video codec.

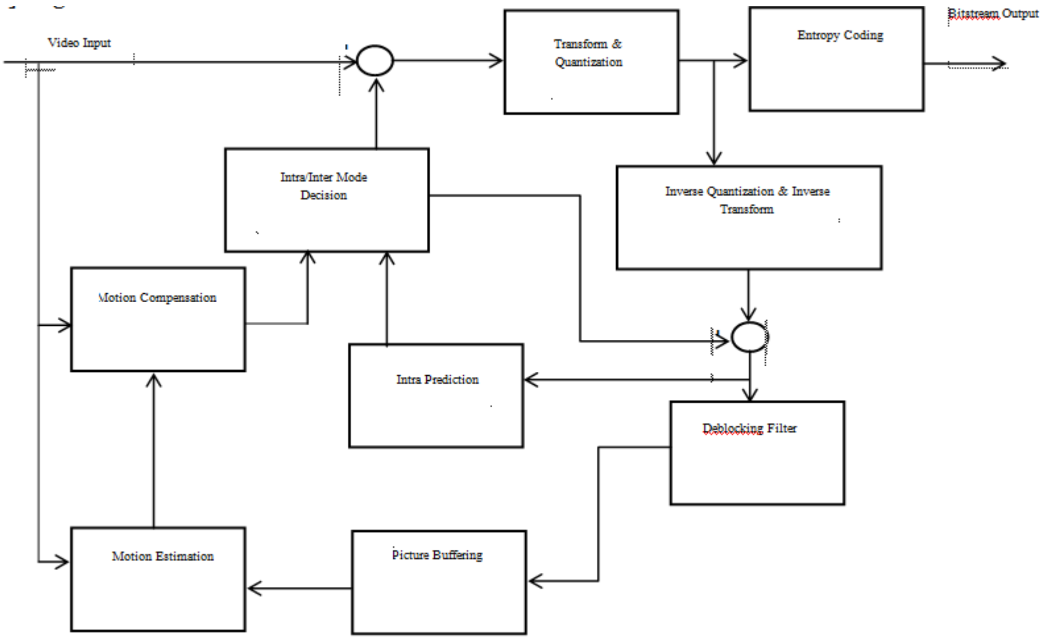

Figure 1: Block Diagram of H264/AVC Encoder

Intra Prediction

Intra coding refers to the case where only spatial redundancies within a video picture are exploited. The resulting frame is referred to as an I-frame. I-frames are typically encoded by directly applying the transform to the different macroblocks (MBs) in the frame. Consequently, encoded I-pictures are large in size since a large amount of information is usually present in the frame, and no temporal information is used as part of the encoding process. In order to increase the efficiency of the intra coding process in H.264/AVC, spatial correlation between adjacent MBs in a given frame is exploited [7]. The idea is based on the observation that adjacent MBs tend to have similar properties. Therefore, as a first step in the encoding process for a given MB, one may predict the MB of interest from the surrounding MBs (typically the ones located on top and to the left of the MB of interest, since those MBs would have already been encoded). The difference between the actual MB and its prediction is then coded, which results in fewer bits to represent the MB of interest as compared to when applying the transform directly to the MB itself.

For the luma samples, the prediction block may be formed for each 4x4 subblock, each 8x8 block, or for a 16x16 macroblock [11]. One case is selected from a total of 9 prediction modes for each 4x4 and 8x8 luma blocks; 4 modes for a 16x16 luma block; and 4 modes for each chroma block.

M	A	B	C	D	E	F	G	H
I	a	b	c	d				
J	e	f	g	h				
K	i	j	k	l				
L	m	n	o	p				

Figure 2: Prediction Samples of A 4x4 Block

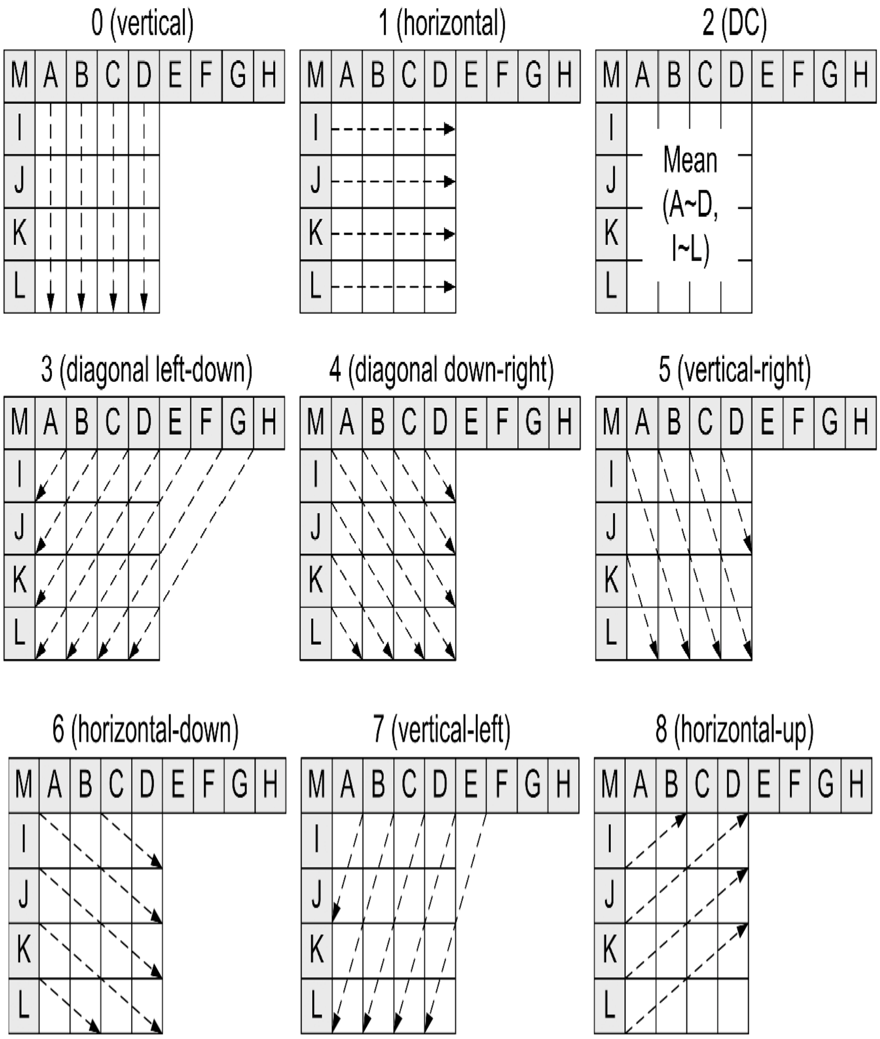

Figure 3: Nine Prediction Mode of A 4x4 Block Intra Prediction [10]

Intra 4× 4 Prediction

For the predicted samples [a, b, . . . , p] of the current block, the above and left previously reconstructed samples [A, B, . . . , M] are used according to direction modes. The arrows in Figure 3 indicate the direction of prediction in each mode[1].

For mode 0 (vertical) and mode 1 (horizontal), the predicted samples are formed by extrapolation from upper samples [A, B, C, D] and from left samples [I, J, K, L],respectively. For mode 2 (DC), all of the predicted samples are formed by mean of upper and left samples [A, B, C, D, I, J, K, L]. For mode 3 (diagonal-down-left), mode 4 (diagonal-down-right), mode 5 (vertical-right), similarly in mode 6 (horizontal-down), mode 7 (vertical-left), and mode 8 (horizontal-up), the predicted samples are formed from a weighted average of the prediction samples A–M[16].

For example, in the case where Mode 3 (Diagonal-Down-Left prediction) is chosen, the values of a to p are given as follows:

a is equal to (A+2B+C+2)/4

b, e are equal to (B+2C+D+2)/4

c, f, i are equal to (C+2D+E+2)/4

d, g, j, m are equal to (D+2E+F+2)/4

h, k, n are equal to (E+2F+G+2)/4

l, o are equal to (F+2G+H+2)/4, and

p is equal to (G+3H+2)/4.

The remaining modes are defined similarly according to the different directions as shown in Figure 3. Note that in some cases, not all of the samples above and to the left are available within the current slice: in order to preserve independent decoding of slices, only samples within the current slice are available for prediction[1]. The encoder may select the prediction mode for each block that minimizes the residual between the block to be encoded and its prediction.

Intra 8 × 8 Prediction

Similar to intra 4x4 block, 8x8 luma block also has 9 prediction mode based on the direction of Figure 3. For prediction of each 8x8 luma block, one mode is selected from the 9 modes, similar to the 4x4 intra-block prediction.

Intra 16 × 16 Prediction

In Intra 16x16 block, there are four prediction modes: Vertical, Horizontal, DC and Plane prediction, which are listed in Figure 4 and Table 1. The 16x16 intra prediction works well in a gently changing area.

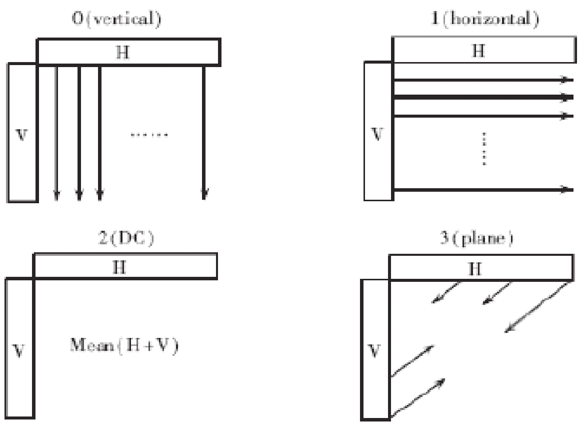

Figure 4: Intra 16 x 16 Prediction Modes [10]

Table 1: Four Intra 16 x 16 Prediction Modes

Mode 0 (Vertical)	Extrapolation From Upper Samples (H)
Mode1 (Horizontal)	Extrapolation from left samples (V)
Mode 2 (DC)	Mean of upper and left-hand samples (H + V)
Mode3 (Plane)	A linear ‘plane’ function is fitted to the upper and left- hand samples H and V. This works well in areas of smoothly-varying luminance.

Intra Chroma Prediction

Each Chroma component of a macroblock is predicted from chroma samples above and/or to the left that have previously been encoded and reconstructed.

The chroma prediction is defined for three possible block sizes, 8x8 chroma in 4:2:0 format, 8x16 chroma in 4:2:2 format, and 16x16 chroma in 4:4:4 format [13]. The 4 prediction modes for all of these cases are very similar to the 16x16 luma prediction modes, except that the order of mode numbers is different: mode 0 (DC), mode 1 (horizontal), mode 2 (vertical), and mode 3 (plane).

Transform, Quantization & CAVLC

The residual macroblock X is transformed using 4x4 Integer Transform. This transform is based on the DCT with some fundamental differences. The 4 x 4 DCT of an input array X is given by

$$Y = A X A^T = \begin{bmatrix} a & a & a & a \\ b & c & -c & -b \\ a & -a & -a & a \\ c & -b & b & -c \end{bmatrix} [X] \begin{bmatrix} a & b & a & c \\ a & c & -a & -b \\ a & -c & -a & b \\ a & -b & a & -c \end{bmatrix}$$

Where

$a = 1/2$, $b = 1/2\cos \pi/8$ and $c = 1/2 \cos(3\pi/8)$ (1)

The forward Transform is performed as

$Y = \otimes X C^T_f E_f$ (2)

The core Transform in Integer transform is given by

$W = C_f X C^T_f$ (3)

The corresponding Inverse Transformation is given by

$X \otimes C^T_i (Y E_i) C_i$ (4)

H.264 uses a scalar quantizer and the quantization incorporates the post and prescaling matrices. The basic forward quantizer operation is as follows:

$Z_{ij} = \text{round}(Y_{ij}/Q_{step})$ (5)

Inverse quantization operation is given by

$Y'_{ij} = Z_{ij} \times Q_{step}$ (6)

In the present work, all the nine modes of Intraprediction have been successfully coded in Matlab.

PROPOSED ALGORITHM

The input image is in TIF format. The algorithm computes PSNR and Compression for different values of Quantization steps with and without Intraprediction.The user has the choice of entering desired Quantization step for the computation. Further, it displays the Menu for the selection of Intraprediction, and the mode of Intraprediction. After the Intraprediciton mode is selected, the prediction matrix is formed from samples in the current frame that has been previously encoded[12]. The prediction P is subtracted from the current macroblock to produce a residual macroblock 'X' of 4x4 size. Transformation, Quantization and their inverses are performed on 'X'.

The Prediction matrix is updated according to the mode of Intra-prediction to process the next macroblock. PSNR is computed for the reconstructed image by computing root mean square error. CAVLC is applied to Y, Cb and Cr components.

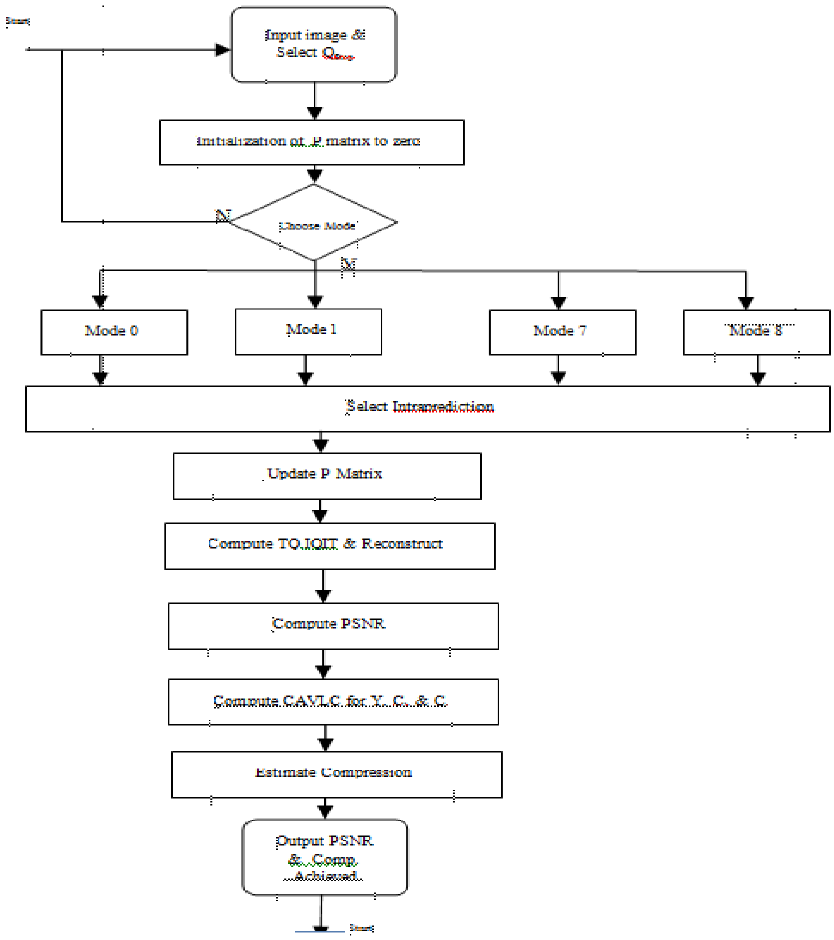

Figure 5: Flowchart of H.264/AVC Encoder Implementing All Intraprediction Modes

SIMULATION RESULTS

Table 2: Simulation Results of Images: PSNR and Compression Achieved With Intraprediction

Intrapre Diction Mode	Blue Hills 800 X 600 Pixels		Clock Geneva 1024 X 768 Pixels		Stefan 512 X 512 Pixels	
Qp = 16	Psnr (Db)	Compre-Ssion Achieved	Psnr (Db)	Compre-Ssion Achieved	Psnr (Db)	Compre-Ssion Achieved
0	34.1	11.7	39.1	29.6	35.7	15.8
1	34.2	12.1	38.6	29.4	35.2	15.1
2	34.1	7.6	38.9	25.8	35.4	10.7
3	34.3	10.8	39.8	27.4	35	14.2
4	32.6	8.4	37	19.4	33.5	10.9
5	32.8	8.6	36.1	19	34	11.7
6	33.1	9.1	38.2	23.4	34.1	12
7	34.5	11.1	40	28.4	35.6	14.3
8	34.6	11	40.4	29.7	36.9	15.9
Without IntraPre-diction	34.8	7.9	40.3	27.8	37.1	11.9

Original Image, 512x512 Pixels

Without Intra Prediction PSNR = 37.1 Db Compression = 11.9 Pixels

With Horizontal Intra Prediction PSNR = 35. 2 Db Compression = 15.1

With Vertical Intra Prediction PSNR = 35. 7 Db Compression = 15. 8

With Horizontal Up Intra Prediction PSNR = 36. 9 Db Compression = 15. 9
Figure 6: Reconstructed Stefan Image Without and with Intra Prediction

CONCLUSIONS

Matlab codes have been developed for H.264 Encoder for all the nine modes of Intraprediction. The simulation results for various resolutions of pictures show better compression of about 25% in all Intraprediction Modes when compared to the Codec without Intraprediction. Of the nine modes of Intraprediction implemented, Vertical Mode, Horizontal Mode and Horizontal up Mode offer the highest compression (about 12 to 30), without sacrificing on the quality of the reconstructed picture (PSNR achieved is of about 34 dB to 40 dB). Presently, work is under progress for the implementation of these modes of Intraprediction on FPGA using the industry standard hardware design language, Verilog.

REFERENCES

1. Manjanaik. N, Manjunath. R " Intra Frame Coding for Advanced Video Coding Standard to Control PSNR and Bitrate Using Gaussian Pulse" International Journal of Emerging Science and Engineering (IJESE) ISSN: 2319–6378, Volume-2, Issue-5, **March 2014.**
2. ITU-T Recommendation H.264 and ISO/IEC 14496-10 (MPEG-4) AVC, "Advanced Video Coding for Generic Audiovisual Services", Version 3: 2005.
3. T. Wiegand, G. J. Sullivan, G. Bjontegaard, and A. Luthra, "Overview of the H.264/AVC video coding standard," IEEE Trans., Circuits Syst. Video Technol., vol. 13, no.7 pp.560-576, July 2003.
4. A M. Tekalp, "Digital Video Processing", Prentice Hall, 1995.
5. C.E. Shannon, "A mathematical theory of communication," Bell System Technical Journal, vol.27, pp. 398-403, July 1948.
6. Y. Wang, J. Ostermann, Y. Q. Zhang, "Video processing and communications", Prentice Hall, New Jersey, 2001.
7. J. Rissanen and K. M. Mohiuddin, "A multiplication-free multialphabet arithmetic code", IEEE Transactions on Communications, Vol. 37, No. 2, pp. 93-98, Feb. 1989.
8. Joint Video Team, Draft ITU-T Recommendation and Final Draft International Standard of joint video specification, ITU-T Recommendation H.264 and ISO/IEC 14496-10 AVC, March 2005.
9. Iain E. G. Richardson, "H. 264 and Video Compression", John Wiley and Sons, 2003.

10. Feng Pan, Xiao Lin, Susanto Rahardja, Keng Pang Lim, Z. G. Li, Dajun Wu, Si Wu, "Fast mode decision algorithm for intraprediction in H.264/AVC Video coding", Circuits and Systems for Video Technology, IEEE Transactions on Volume 15, July 2005.

11. S. Kwon, A. Tamhankar, K. R. Rao, "Overview of H. 264/MPEG-4 Part 10", IEEE Transactions on Circuits and Systems for Video Technology, 2003.

12. Thomas Weigand, Gary J. Sullivan Gisle Bjonte gaard, and Ajay Luthra, "Over view of the H.264/AVC Video Coding Standard". IEEE Transactions on Circuits and Systems for Video Technology, July 2003.

13. Gulistan Raja, Sadiqullah Khan, Muhammad Javed Mirza, "VLSI architecture and implementation of H.264 integer transform," The IEEE 2005 Workshop on Signal Processing Systems (SIPS'05), 2005.

14. N. Keshaveni, S. Ramachandran, K.S. Gurumurthy, "Design and Implementation of Integer Transform and Quantization Processor for H.264 Encoder on FPGA", International Conference on Advances in Computing,

15. Control and Telecommunication Technologies, December 2009.

16. N. Keshaveni, S. Ramachandran, K.S. Gurumurthy, "Implementation of Context Adaptive Variable Length Coder for H.264 Video Encoder", International Journal of Recent Trends in Engineering [ISSN: 1797-9617] by the Academy Publishers, Finland, Jan. 2010.

17. Draft ITU-T Recommendation and Final Draft International Standard of Joint Video Specification (ITU-T Rec. H.264/ISO/ IEC 14496-10 AVC), Mar. 2003.

18. First Author: Imran Ullah Khan obtained B.Tech Degree (with Honors in Electronics and Communication Engineering) and M.Tech (Communication and information systems) from Aligarh Muslim University, Aligarh. Pursuing Ph.D. from Mewar University, Chittorgarh .

19. Mail ID: imranuk79@gmail.com

20. Second Author: Dr. M.A.Ansari received the B.Tech Degree from Aligarh Muslim University, Aligarh, the M.Tech degree from the I.I.T. Roorkee in 2001 and Ph.D. degree from the I.I.T. Roorkee in 2009. He is presently working in Islamic University, Madinah, KSA. He is also the senior member of IEEE.

21. Mail ID: ma.ansari@ieee.org