International Journal of Computer Sciences and Engineering Open Access
Review Paper Vol.-6, Issue-11, Nov 2018 E-ISSN: 2347-2693

A Systematic Review on Real Time Video Compression and Enhancing Quality Using Fuzzy Logic

Upendra Kumar Srivastava[1*], Navin Prakash[2]

[1]Dept. of Computer Science and Engineering I.F.T.M. University, Moradabad ,India
[2]Dept. of Computer Science and Engineering I.F.T.M. University, Moradabad ,India

Corresponding Author: upendra_srivas@rediffmail.com, Tel.: +91-7652073033

Available online at: www.ijcseonline.org

Accepted: 26/Nov/2018, Published: 30/Nov/2018

***Abstract*—** This paper provides the critical reviews on Real time Video Compression and Efficient use of Fuzzy Logic Techniques used in Video Compression and Quality Enhancement. Since the Internet is highly heterogeneous environment video codec needs to be able to generate bit streams that are highly scalable in terms of bandwidth and processing requirements looking all these problems this research paper explores the possibility of better compression ratio in real time and quality enhancement by efficient use of fuzzy logic . The **first section** of this paper tells the overview of the real time video compression .The **second section** of this paper describes the related work which has been done in the past regarding real time video compression it consists a **Table-1** in reference of the time line of real time video compression and **Table -2** about the differences between H.265 and H.264 .The **third section** of this paper consists a **Table-3** which represents about the research time line using fuzzy logic in video compression .The **fourth section** of this paper consists a **Table-4** which represents the research time line of real time video compression. Finally the conclusion of this paper is an overview on past, present and future trends in Video Compression Technologies, review of the improvements and development in video encoding over the last two decades with future possibilities.

***Keywords*-**Real time, Video compression, Fuzzy logic, Motion vector estimation, Bit rate

I. Introduction

An important and difficult problem is real time video compression which attracts to the researcher to deal with it . However the evolution is going on and day by day we encounter with new technologies and video codec standard as H.264/AVC and video H.265/HEVC but the cost of these algorithm and compatible hardware is a big question also . this research paper explores the possibility of modified real time video compression algorithm and quality enhancement by fuzzy logic techniques. This paper has been divided into three sections the first section describes the time line about the real time video compression the second section describes the time line of real time video compression and the third section describes the time line about the use of fuzzy logic techniques to improve the quality of video compression .

Need of compression: According to the latest **Cisco Visual Networking Index (VNI)** Complete Forecast,

there will be nearly **1.9 billion** Internet video users by **2021**, up from **1.4 billion** in **2016**. The world will be watching **3 trillion** minutes of Internet video per month by **2021**, which is **5 million** years of video per month. And video will continue to dominate overall Internet traffic – representing **80%** of all Internet traffic by **2021**, up from **67% in 2016**. This prediction shows that in near future there is a need of higher coding efficiency, higher through put and low power.

A simple uncompressed video file will contain an array of 2D buffers containing pixel data for each frame. So it's a 3D (2 spatial dimensions and 1 temporal) array of bytes. Each pixel takes 3 bytes to store , one byte each for the three primary colors (red, green and blue). 1080p @ 60 Hz = 1920x1080x60x3 = 370 MB/sec of raw data. This is next to impossible to deal with. A 50GB Blu-ray disk will only hold 2 minutes.

II. RELATED WORK

The video calling services are not exchanging files, so no file format is used. Each implements a streaming protocol that specifies how frames are structured, buffered, synced, resolution negotiated, etc. Apple's Face Time uses a protocol that uses an **h.264 or h.265** transport stream for the video portion and proprietary features for the content negotiation.

Storage Area Network (SAN) A SAN consists of interconnected hosts, switches and storage devices. The components can be connected using a variety of protocols.

Fibre Channel is the original transport protocol of choice. Another option is Fibre Channel over Ethernet (FCoE), which lets organizations move Fibre Channel traffic across existing high-speed Ethernet, converging storage and IP protocols onto a single infrastructure. Other options include Internet Small Computing System Interface (iSCSI), commonly used in small and midsize organizations, and InfiniBand, commonly used in high-performance computing environments.

Network-attached storage (**NAS**) is a file-level computer data storage server connected to a computer network providing data access to a heterogeneous group of clients. NAS is specialized for serving files either by its hardware, software, or configuration. It is often manufactured as a computer appliance – a purpose-built specialized computer. NAS systems are networked appliances which contain one or more storage drives, often arranged into logical, redundant storage containers or RAID. Network-attached storage removes the responsibility of file serving from other servers on the network. They typically provide access to files using network file sharing protocols such as NFS, SMB/CIFS, or AFP. From the mid-1990s, NAS devices began gaining popularity as a convenient method of sharing files among multiple computers. Potential benefits of dedicated network-attached storage, compared to general-purpose servers also serving files, include faster data access, easier administration, and simple configuration.

SAN is associated with structured workloads such as databases, while **NAS** is generally associated with unstructured data such as **video and medical images**.

Real time compression is an inline storage optimization technology often implemented on an appliance that is commonly deployed into NAS environment. The appliance sits in front of the NAS head , processing all data coming into and out of the NAS through the real –time compression technology .This produces a minimum 50% savings in capacity upfront and a ripple effect throughput the other storage tiers that can be several times that amount .

Table 1: Time line of real time video compression

Video Coding Standards	Year of Development	Publisher	Primary Application	Bit Rate
H.120	**1984** with a revision in 1988	CCITT (now ITU-T) International Telecommunication Union	Video Telephony and teleconferencing over ISDN	64 kbit/s.
H.261 first member of the H.26x	**Nov (1988-1990)**	ITU-T9 (Video Coding Experts Group (VCEG))	Video Telephony and Video conferencing use over ISDN as part of the H.320 protocol suite.	Multiples of 64Kbit/s (“p x 64”)
H.262/MPEG-2 Part 2	**1995**	Joint partnership between VCEG and MPEG,	DVD Video , Blu-Ray, Digital Video Broadcasting , SVCD	
H.263 Based on experience from H.261, and the MPEG-1 and MPEG-2 standards.	**1996**	ITU-T9 (Video Coding Experts Group (VCEG))	Video Telephony , Video Conferencing adopted ITU-T H.324 (PSTN) , H.310 (B-ISDN and Video on Mobiles Phones (3GP)	33.6 kbit/s and up
H.263+		ITU-T9 (Video Coding Experts Group (VCEG))	Second version of H.263 with enhanced features like improved compression performance, bit stream scalability,	33.6 kbit/s and up
H.263v3 (also known as H.263++ or H.263 2000 and **MPEG-4 Part 2**	First edition release date **1999** and last amendment third edition **2009**	ISO/IEC 14496-2(1999) ISO/IEC 14496-2(2009)	SIMPLE PROFILE (SP) some low end video conferencing systems, electronic surveillance systems	
H.264 or MPEG-4 AVC (or MPEG-4 Part 10). Extended from H.263, MPEG-4 Visual	Version 1 (Edition 1): (May 30, **2003** Version 25 (Edition 12): (April 13, **2017**	ISO/IEC 14496–10, ITU-T H.264 Developed by	formats for the recording, compression, and distribution of video content. It supports resolutions up to 4096×2304, including 4K UHD	Variable bit rate
H.264/AVC (Advanced Video Coding)	Version 1 (Edition 1): (May 30, **2003** Version 25 (Edition 12): (April 13, **2017**	ITU-T (VCEG) with the ISO/IEC JTC1 (MPEG). as the Joint Video Team (JVT) formally, ISO/IEC 14496-10– MPEG-4 Part 10, (AVC)	Best known as being one of the video encoding standards for Blu-ray Discs. Widely used by streaming internet sources,	Variable bit rate
H.265/ HEVC (High Efficiency Video Coding) and MPEG-H Part 2 is a successor to H.264/MPEG-4 AVC	**2013**	ITU-T (VCEG) with the ISO/IEC JTC1 (MPEG). as the Joint Video Team (JVT) formally, ISO/IEC 14496-10– MPEG-4 Part 10, (AVC)	How video data is displayed, either online, on television and even in the surveillance industry	No standard bit rate

Time Line Description

H.264/AVC H.264/MPEG4-AVC is the video coding standard of the ITU-T Video Coding Experts Group (VCEG) and the ISO/IEC Moving Picture Experts Group (MPEG).

H.264/MPEG4-AVC has become the most widely accepted video coding standard since the deployment of MPEG2 at the dawn of digital television, and it may soon overtake MPEG2 in common use. It covers all common video applications ranging from mobile services and **videoconferencing** to IPTV, HDTV, and HD video storage 50% higher coding efficiency than MPEG---2 (used in DVD, US terrestrial broadcast) It has efficient motion compensation and reduced bit-rate. Different block sizes are used for performing motion compensation which results in better video quality.

The basic processing unit is 16x16 pixel macro blocks. The two entropy encoding methods used are CAVLC and CABAC. For all syntax elements, Context-Adaptive Variable Length Coding (CAVLC) uses a single codeword set. Run Length encoding is used to code the transformation coefficient. In Context-Adaptive Binary Arithmetic coding, statistics of previously coded symbols are used for encoding and it uses arithmetic coding for transmission. HDTV broadcasting, Internet Video, Video Conference etc are some of the applications of H.264/MPEG-4 AVC.

The ITU approved the new H.264 standard in May 2003. The ISO approved the standard in October of 2003 as MPEG-4 Part 10, Advanced Video Coding or AVC. The 2x improvement offered by

H.264 creates new market opportunity such as the following possibilities:
• VHS-quality video at about 600 Kbps. This can enable video delivery on demand over ADSL lines.
• An HD movie can fit on one ordinary DVD instead of requiring new laser optics.

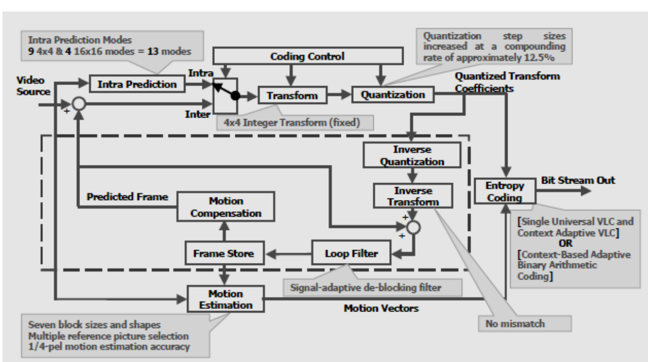


Figure:1 Block Diagram of H.264/AVC

While H.264 uses the same general coding techniques as previous standards, it has many new features that distinguish it from previous standards and combine to enable improved coding efficiency. The main differences are summarized in the encoder block diagram in Figure 1 and described briefly below:

Intra Prediction and Coding: When using intra coding, intra prediction attempts to predict the current block from

the neighboring pixels in adjacent blocks in a defined set of directions. The difference between the block and the best resulting prediction is then coded rather than actual block. This results in a significant improvement in intra coding efficiency.

Inter Prediction and Coding: Inter-frame coding in H.264 leverages most of the key features in earlier standards and adds both flexibility and functionality including various block sizes for motion compensation, quarter-pel motion compensation, multiple-reference frames, and adaptive loop deblocking.

Block sizes: Motion compensation can be performed using a number of different block sizes. Individual motion vectors can be transmitted for blocks as small as 4x4, so up to 32 motion vectors may be transmitted for a single macroblock in the case bi-directional prediction. Block sizes of 16x8, 8x16, 8x8, 8x4, and 4x8 are also supported. The option for smaller motion compensation improves the ability to handle fine motion detail and results in better subjective quality including the absence of large blocking artifacts.

Quarter-Pel Motion Estimation: Motion compensation is improved by allowing half-pel and quarter-pel motion vector resolution.

Multiple Reference Picture Selection: Up to five different reference frames can be used for inter-picture coding resulting in better subjective video quality and more efficient coding. Providing multiple reference frames can also help make the H.264 bitstream more error resilient. Note that this feature leads to increased memory requirement for both the encoder and the decoder since multiple reference frames must be maintained in memory.

Adaptive Loop Deblocking Filter: H.264 uses an adaptive deblocking filter that operates on the horizontal and vertical block edges within the prediction loop to remove artifacts caused by block prediction errors. The filtering is generally based on 4x4 block boundaries, in which two pixels on either side of the boundary may be updated using a 3-tap filter. The rules for applying the loop deblocking filter are intricate and quite complex.

Integer Transform: H.264 employs a purely integer 4x4 spatial transform which is an approximation of the DCT instead of a floating-point 8x8 DCT. Previous standards had to define rounding-error tolerances for fixed point implementations of the inverse transform. Drift caused by mismatches in the IDCT precision between the encoder and decoder were a source of quality loss. The small 4x4 shape helps reduce blocking and ringing artifacts.

Quantization and Transform Coefficient Scanning: Transform coefficients are quantized using scalar quantization with no widened dead-zone. Thirty-two different quantization step sizes can be chosen on a macro block basis similar to prior standards but the step sizes are increased at a compounding rate of approximately 12.5%, rather than by a constant increment. The fidelity of chrominance components is improved by using finer quantization step sizes compared to luminance coefficients, particularly when the luminance coefficients are coarsely quantized.

Entropy Coding: The baseline profile uses a Universal VLC (UVLC)/Context Adaptive VLC (CAVLC) combination and the main profile also supports a new Context-Adaptive Binary Arithmetic Coder (CABAC). **UVLC/CAVLC**: Unlike previous standards that offered a number of static VLC tables depending on the type of data under consideration, H.264 uses a Context-Adaptive VLC for the transform coefficients and a single Universal VLC approach for all the other symbols. The CAVLC is superior to previous VLC implementations but without the full cost of CABAC.

Context-Based Adaptive Binary Arithmetic Coding (CABAC): Arithmetic coding uses a probability model to encode and decode the syntax elements such as transform coefficients and motion vectors. To increase the coding efficiency of arithmetic coding, the underlying probability model is adapted to the changing statistics within a video frame, through a process called context modeling. Context modeling provides estimates of conditional probabilities of the coding symbols. Utilizing suitable context models, the given inter-symbol redundancy can be exploited by switching between different probability models, according to already coded symbols in the neighborhood of the current symbol. Each syntax element maintains a different model (for example, motion vectors and transform coefficients have different models). CABAC can provide up to about 10% bitrate improvement over UVLC/CAVLC.

H.264 supports three profiles: baseline, main, and extended. Currently the baseline profile and main profile are generating the most interest. The baseline profile requires less computation and system memory and is optimized for low latency. It does not include B frames due to its inherent latency and CABAC due to the computational complexity. The baseline profile is a good match for video telephony applications as well as other applications that require cost-effective real-time encoding .

The main profile provides the highest compression but requires significantly more processing than the baseline profile making it difficult for low-cost real-time encoding and also low-latency applications. Broadcast and content storage applications are primarily interested in the main profile to leverage the highest possible video quality at the lowest bitrate

H.264 main profile is the most compute intensive of the current standards. It is very attractive though due to the high compression efficiency. Since many of the coding tools are new, fixed function video decoder devices cannot support it without developing new optimized silicon. The DM642 media processor supports real-time main profile decoding at D1 resolution and is finding early market acceptance for products such as IP set-top boxes.
H.264 baseline profile is targeted for video telephony and requires significantly lower performance than the main profile. This is partly a function of the tools and secondly the more relaxed requirements for motion estimation and overall quality versus broadcast content. The DM642 can support simultaneous encoding and decoding of H.264 baseline at up to VGA resolution.

H.265/HEVC (High Efficiency Video Coding)

High Efficiency Video Coding (HEVC) [11] is a video compression standard developed by a Joint Collaborative Team on Video Coding (JCT-VC) of ITU-T and ISO/IEC 23008-2. HEVC is a successor to H.264/MPEG-4 AVC and an evolution of the existing video coding recommendations (ITU-T H.261, ITU-T H.262, ITU-T H.263 and ITU-T H.264). H.265 was developed in response to the growing need for higher compression of motion picture for various ICT solutions like internet streaming, communication, videoconferencing, digital storage media and television broadcasting [5].The technical content
specifications were officially finalized on April 13, 2013. The H.265 standard was designed to be applicable for almost all existing H.264 applications while putting emphasize on high resolution video HEVC particularly focuses on two key issues (1)increased video resolution (2) increased use of parallel processing architecture . HEVC"s key benefit is the reduction of bandwidth bitrate up to 50% while maintaining the same video quality.

High Efficiency Video Coding (HEVC), or **H.265**, is a video compression standard designed to substantially improve coding efficiency when compared to its precedent, the Advanced Video Coding (AVC), or H.264. With an increasing growth of video streaming on the Internet. You can get the same quality at a lower bit rate compared to H.264, it is not 2-to-1 savings, it is more 1.5-to-1, practically it is not half it may be from 33 percent or more, and results can vary based on what kind of content we are sending when it is compared to H.264/AVC

Over popular websites such as Netflix and YouTube, and with 4K cameras gaining new ground in the market, a considerable amount of storage and bandwidth is required. HEVC promises a 50% storage reduction as its algorithm uses efficient coding by encoding video at the lowest possible bit rate while maintaining a high image quality level.

With this new format, image resolutions around 8192×4320 become possible to display and stream. To demonstrate the incredible power of this codec, a subjective video performance study was made between these two codecs to understand how intensely is this bit reduction. The study showed the bit reduction is inversely proportional to the video image quality, where HEVC/H.265 presented a bit reduction of 52% at 480p and 64% at 4K UHD when compared to H.264. Besides this outstanding bit reduction, when compared to H.264, HEVC/H.265 delivers a significantly better visual quality, when compressed to the same file size or bitrate.

Table 2 :Dfferences between H.265 (HEVC) and H.264 (MPEG 4 AVC)

Category	**H.264**	**H.265**
Names	MPEG 4 Part 10 AVC (Introduced in 2004)	MPEG-H, HEVC, Part 2 (Approved in Jan 2013)
Key	- 40-50% bit rate 40 to 50%	- 40-50% the bit rate reduction at the same visual quality
Improvement	reduction compared to MPEG-2 - Led the growth of HD content delivery for Broadcast and Online	compared to H.264 - Potential to realize UHD, 2K, 4K for Broadcast and Online (OTT)
Industry adoption	Dominant and accepted video codec for Terrestrial, Cable, Satellite and IPTV broadcast. (ATSC/DVB/ISDB) Widely used across Blu-Ray, security systems, videoconferencing , mobile video, media players, video chat etc	Implementation demonstration across NAB, IBC and other events starting 2012 from companies e.g. ATEME, Broadcom, Thomson , Harmonic (Cisco), Ericsson, Qualcomm etc.. Increased R&D across Encoder/Decoder /CE vendors for software and hardware based solutions
Specification	Support Up to 4K (4,096×2,304) Supports up to 59.94 fps 21 profiles ; 17 levels	Up to 8K UHDTV (8192×4320) Supports up to 300 fps 3 approved profiles, draft for additional 5 ; 13 levels
Progression	Successor to MPEG-2 Part	Successor to MPEG 4 AVC, H.264
Drawbacks	Unrealistic for UHD content delivery due to high bit rate requirements. Frame rate support restricted to 59.94	Computationally expensive (~ 300 % +) due to larger prediction units and expensive Motion Estimation (Intra prediction with more nodes, asymmetric partitions in Inter Prediction).
Compression Model	Hybrid spatial-temporal prediction model - Flexible partition of Macro Block (MB) , sub MB for motion estimation - Intra Prediction (extrapolate already decoded neighboring pixels for prediction) - Introduced multi-view extension 9 directional modes for intra prediction - Macro Blocks structure with maximum size of 16x16 - Entropy coding is CABAC and CAVLC	Enhanced Hybrid spatial-temporal prediction model - Flexible partitioning, introduces Coding Tree Units (Coding, Prediction and Transform Units -CU, PU, TU) - 35 directional modes for intra prediction - Superior parallel processing architecture, enhancements in multi-view coding extension - CTU supporting larger block structure (64x64) with more variable sub partition structures - Entropy coding is only CABAC

Compared to H.264, H.265 offers up to twice the data compression with the same level of video quality. It is designed to support future resolutions up to 8K UHD (8192x4320) compared to the 4K (4092x2160) that the H.264 supports. Some newer equipment such as TV's are starting to ship with a built in hardware decoder to play H.265 content, but the improved quality and reduced bandwidth comes at a cost. H.265 encoding and decoding requires much more processing power over H.264 so the cost of H.265 solutions are still significantly higher.

Cost of HEVC : With two HEVC patent pools, HEVC Advance and MPEG LA, the adoption cost of HEVC is expected to be roughly ***16 times more expensive*** per unit than its predecessor H.264. Interestingly Sony, Panasonic, Qualcomm, Nokia and Broadcom with loads of patents are still not part of the current two pools.

1. Larger screen resolution will requires higher frame rates 60- 120 fps from current 24-30 fps, given the increase in fps, what overall efficiency savings could HEVC realistically achieve?

2. Given state of bandwidth networks, growth in video coupled with multi screen delivery and absence of UHD content, will encoding existing SD/HD content with bitrate efficiency become more likely business case for HEVC rather than 4K?

3. How far is HEVC adoption from broadcast industry standard specifications (DVB/ATSC) given footprint of legacy equipment and transmission infrastructure?
4. What are the sector specific services which could lead the transition and embrace HEVC sooner that rest of the pack – mobile video services, OTT players? Similarly role of international events like 2014 FIFA world cup and Rio 2016 Olympics towards selective adoption?

Versions of HEVC

On January 25, 2013, the ITU announced that HEVC had received first stage approval (consent) in the ITU-T Alternative Approval Process (AAP). On the same day, MPEG announced that HEVC had been promoted to Final Draft International Standard (FDIS) status in the MPEG standardization process.
On April 13, 2013, HEVC/H.265 was approved as an ITU-T standard. The standard was formally published by the ITU-T on June 7, 2013 and by the ISO/IEC on November 25, 2013.
On July 11, 2014, MPEG announced that the 2nd edition of HEVC will contain three recently completed extensions which are the multiview extensions (MV-HEVC), the range extensions (RExt), and the scalability extensions (SHVC).
On October 29, 2014, HEVC/H.265 version 2 was approved as an ITU-T standard. It was then formally published on January 12, 2015.
On April 29, 2015, HEVC/H.265 version 3 was approved as an ITU-T standard.
On June 3, 2016, HEVC/H.265 version 4 was consented in the ITU-T and was not approved during a vote in October 2016.
On December 22, 2016, HEVC/H.265 version 4 was approved as an ITU-T standard.

III. FUZZY LOGIC TECHNIQUES USED IN VIDEO COMPRESSION OR QUALITY ENHANCEMENT

Table : 3 Previous Research work related to Efficient use of fuzzy Logic Techniques used in Video Compression or Quality Enhancement

S.N.	Author/Researcher	Reference	Year and Name of journal	Topic name	Research focused on	Remark
1	Danny H. K. Tsang, Brahim Bensaou, and Shirley T. C. Lam	1	1998 IEEE	Fuzzy-Based Rate Control for Real-Time MPEG Video	Input and output rates of the shaper buffer are controlled by two fuzzy logic-based controllers.	**Fuzzy** logic control is used For bit rate controller
2	Y.-S. Saw P.M. Grant J.M.Hannah	2	1998 IEEE	Quality-optimized MPEG2 video data rate control using fuzzy logic techniques	To improve the rate control Mechanism for a MPEG2 video encoder Fuzzy logic control has been employed .	**Fuzzy** logic control is used For bit rate controller
3	Yun-Teng Roan and Pei-Yin Chen	3	2000 IEEE	A Fuzzy Search Algorithm for the Estimation of Motion Vectors	Using the inter-block and inter-frame fuzzy prediction, motion vectors of image blocks are determined	**Fuzzy** technique is used for Block motion estimation
4	Ming-Chieh Chi, Mei-Juan Chen and Ching-Ting Hsu	4	2004 IEEE	Region-of-interest video coding by fuzzy control for h.263+ standard	Based on TMN8 utilizes fuzzy logic control assigns	**Fuzzy** technique is used For bit rate controller

					weighted factor to each Macro block (MB) and allocate different bits to each MB	As ROI
5	Pei-Jun Lee and Ming-Long Lin	5	2006 ETRL Journal	Fuzzy Logic Based Temporal Error Concealment for H.264 Video	By fuzzy logic the lost block is determine by less computation time in H.264 standard.	**Fuzzy** technique is used For error concealment
6	Mehdi Rezaei, Miska M. Hannuksela, and Moncef Gabbouj,	6	2008 IEEE	Semi-Fuzzy Rate Controller for Variable Bit Rate Video	VBR video bit stream controlling the quantization parameter (QP) is controlled by a fuzzy rate controller	**Fuzzy** technique is used For bit rate controller
7	WU Jing, DU Xin, ZHU Yun-fang, GU Wei-kang	7	2008 IEEE	Adaptive Fuzzy Filter Algorithm for Real-time Video Denoising	An adaptive fuzzy filter algorithm which removes the real time Gaussian noise by spatiotemporal neighbor information through noise estimation motion detection and spatiotemporal neighbor similarity measurement.	**Fuzzy** technique is used for real time Gaussian noise removal
8	S.M.R. Soroushmehr S. Samavi M. Saraee	8	2009 IEEE	Fuzzy block matching motion estimation for video compression	For motion estimation a spatiotemporal fuzzy search algorithm has been proposed to make the search time shorten	**Fuzzy** technique is used for motion vector estimation
9	Junhua Chai, Jun Ying and Li Li	9	2009 IEEE	A Fuzzy Video Pre-filtering Method for Impulse Noise Reduction	For the real time operation by fuzzy detection and fuzzy filtering a non linear filter is designed for reducing video noise	**Fuzzy** technique is used for impulse noise reduction
10	C. Solana-Cipres, L. Rodriguez-Benitez1 J. Moreno-Garcia2 L. Jimenez1 G. Fernandez-Escribano	10	2009 IFSA-EUSFLAT	Real-time segmentation of moving objects in H.264 compressed domain with dynamic design of fuzzy sets	A real time segmentation algorithm on H.264 domain which uses fuzzy logic to describe position , velocity, and size of detected regions of moving objects	By Dynamic design of **Fuzzy** sets In H.264 compression domain in real time segmentation can be achieved
11	Zhan Xuefeng Zhu Xiuchang	11	2010 journal of electronics (china)	A novel temporal error concealment method based on Fuzzy reasoning for H.264	Temporal EC (TEC). On TEC techniques, temporal correlation between successive frames of video	**Fuzzy** logic technique for error concealment Applicable to H.264/AVC

					sequences is exploited to recover the corrupted MB.	
12	Suvojit Acharjee Sheli Sinha Chaudhuri	12	2012 I.J. Image, Graphics and Signal Processin g	Fuzzy Logic Based Four Step Search Algorithm for Motion Vector Estimation	To reduce the computational complexity of Full Search algorithm the fuzzy logic based Four Step Search algorithm	**Fuzzy** logic technique for motion vector estimation

Fuzzy logic technique has been applied till now **Bit rate control** , **Motion Estimation and Compensation** ,**Error concealment** and **Noise removal** these terms are significant to real time video compression and quality enhancement

Bit rate control

7 different standard widescreen resolutions set of resolutions, bitrates and settings used for high-quality H.264 video encoding, and the reasoning behind those choices.

240p (424x240, 0.10 megapixels)
360p (640x360, 0.23 megapixels)
432p (768x432, 0.33 megapixels)
480p (848x480, 0.41 megapixels, "SD" or "NTSC widescreen")
576p (1024x576, 0.59 megapixels, "PAL widescreen")
720p (1280x720, 0.92 megapixels, "HD")
1080p (1920x1080, 2.07 megapixels, "Full HD")

For each resolution, we use a **bit rate which is the lowest sensible 64% cut** (80% of 80%) of a common Internet link speed (see below) **that still achieves "very good" visual quality**, with no major visible compression artifacts.

Variable bit rate (**VBR**) is a term used in telecommunications and computing that relates to the bit rate used in sound or video encoding. As opposed to constant bit rate (CBR), VBR files vary the amount of output data per time segment. VBR allows a higher bit rate (and therefore more storage space) to be allocated to the more complex segments of media files while less space is allocated to less complex segments. The average of these rates can be calculated to produce an average bit rate for the file.

Opus, Vorbis, MP3, WMA and AAC audio files can optionally be encoded in VBR.[1][2][3] Variable bit rate encoding is also commonly used on MPEG-2 video, MPEG-4 Part 2 video (Xvid, DivX, etc.), MPEG-4 Part 10/H.264 video, Theora, Dirac and other video compression formats. Additionally, variable rate encoding is inherent in lossless compression schemes such as FLAC and Apple Lossless.

Motion Estimation and Compensation

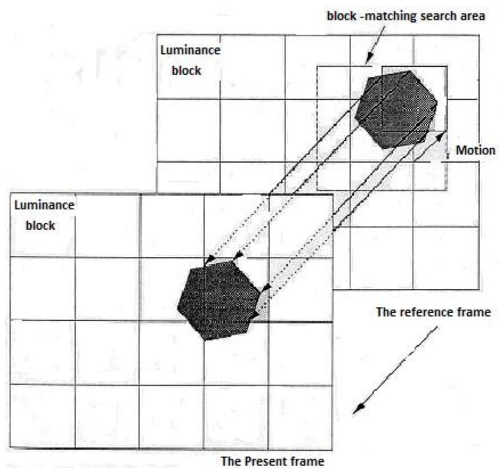
Motion estimation examines the movement of objects in an image sequence to try to obtain vectors representing the estimated motion. Motion compensation uses the knowledge of object motion so obtained to achieve data compression. In inter frame coding, motion estimation and compensation have become powerful techniques to eliminate the temporal redundancy due to high correlation between consecutive frames. In real video scenes, motion can be a complex combination of translation and rotation. Such motion is difficult to estimate and may require large amounts of processing. However, translational motion is easily estimated and has been used successfully for motion compensated coding.

Most of the motion estimation algorithms make the following assumptions:

1.Objects move in translation in a plane that is parallel to the camera plane, i.e., the effects of camera zoom, and object rotations are not considered.
2.Illumination is spatially and temporally uniform.
3.Occlusion of one object by another, and uncovered background are neglected.

There are two mainstream techniques of motion estimation: pel-recursive algorithm (PRA) and block-matching algorithm (BMA). PRAs are iterative refining of motion estimation for individual pels by gradient methods. BMAs assume that all the pels within a block has the same motion activity. BMAs estimate motion on the basis of rectangular blocks and produce one motion vector for each block. PRAs involve more computational complexity and less regularity, so they are difficult to realize in hardware. In general, BMAs are more suitable for a simple hardware realization because of their regularity and simplicity

Figure 2 illustrates a process of block-matching algorithm. In a typical BMA, each frame is divided into blocks, each of which consists of luminance and chrominance blocks. Usually, for coding efficiency, motion estimation is performed only on the luminance block. Each luminance block in the present frame is matched against candidate blocks in a search area on the reference frame. These candidate blocks are just the displaced versions of original block. The best (lowest distortion, i.e., most matched) candidate block is found and its displacement (motion vector) is recorded. In a typical inter frame coder, the input frame is subtracted from the prediction of the reference frame. Consequently the motion vector and the resulting error can be transmitted instead of the original luminance block; thus inter frame redundancy is removed and data compression is achieved. At receiver end, the decoder builds the frame difference signal from the received data and adds it to the reconstructed reference frames. The summation gives an exact replica of the current frame. The better the prediction the smaller the error signal and hence the transmission bit rate.



Error concealment

Live video transmission needs to have very strict transmission delays (cannot afford many retransmissions - ARQ). On the other hand, video transmission is growing even more popular via mobile phones and over the internet, which use noisy channels for the transmission. To solve this problem, a set of techniques were developed, whose purpose was to minimize the influence of the transmission errors at the decoder, taking in consideration the characteristics of the video signal. These techniques are called Error Concealment Techniques and can be divided according to the element, of the transmission system, that has the major part in its implementation [Wang and Zhu 1998]:

• Forward Error Concealment – performed by the encoder;

Figure 2:Process of block Matching Algorithm

• Post processing Error Concealment – performed by the decoder;

• Interactive Error Concealment – performed jointly by the encoder and decoder.

In fact, Forward Error Concealment and Interactive Error Concealment techniques can be viewed has an extension, that takes in consideration the specificities of the video transmission, of the traditional FEC and ARQ error control techniques, respectively [Wang et al. 2002]. In this sense, Post processing Error Concealment techniques are the ones that brought a new way of analyzing this kind of problem. Furthermore, the change of the coding structure, from a pixel based for a object based oriented, implied the birth of Error Concealment Techniques that were also object based oriented (the so called 2^{nd} generation error concealment techniques [Chen and Chen 2002]).

Noise removal

Video noise usually looks like a slight snowyness on video, and it can also look like little flickering blue and green and red and colored pixels . The denoising function in digital imaging devices must consider resource consumption and real-time capability in addition to effective noise-removal performance. One commonly used denoising method is pixel similarity weighted frame averaging (PSWFA).

The fuzzy systems filter noise and improve motion estimation and compensation. Adaptive fuzzy systems further improve the compensation accuracy. Fuzzy systems of if-then rules can filter impulsive noise from signals. An additive fuzzy system learns ellipsoidal fuzzy rule patches from a new pseudo-co variation measure of alpha-stable co variation.

A fuzzy system can also estimate motion vectors and increase the compensation accuracy for video compression. The fuzzy system uses the temporal correlation of the motion field to estimate the motion vectors. First and second order statistics of the motion vectors give ellipsoidal search windows.

IV. RESEARCH WORK RELATED TO REAL TIME VIDEO COMPRESSION

Table-4: Previous Research work related to Real time Video compression

S.N.	Author/Researcher	Reference	Year and Name of journal	Topic name	Research focused on	Remark
1	T. Fryza	13	2008 IEEE	Introduction to Implementation of Real Time Video Compression Method	By using 3-D DCT video signal are compressed	Enhanced DCT compression is used
2	J. C. Galan-Hernandez, V. Alarcon-Aquino, O. Starostenko, J. M. Ramirez-Cortes	14	2010 IEEE	Wavelet-Based Foveated Compression Algorithm for Real-Time Video Processing	Wavelet foveated compression can be used in real-time video processing	if the signal is stationary in the frequency domain, there is no advantage

					frameworks for reducing the communication overhead.	to DWT, and more effort to compute the values.
3	Gary J. Sullivan, Jens-Rainer Ohm, Woo-Jin Han, Thomas Wiegand	15	2012 IEEE	Overview of the High Efficiency Video Coding (HEVC) Standard	**H.265**, is a video compression standard designed to substantially improve coding efficiency when compared to its precedent, the Advanced Video Coding (AVC), or H.264. With an increasing growth of video streaming on the Internet.	Practically it is not half it may be from 33 percent or more, and results can vary based on what kind of content we are sending when it is compared to H.264/AVC
4	Stamos Katsigiannis, Georgios Papaioannou, and Dimitris Maroulis	16	2013 IEEE	A GPU based real-time video compression method for video conferencing	By Using contourlet transformation and Graphics Processing Units	Managing GPU core can be problematic because the software architecture is still under development. New GPUs are coming with new features.
5	Ali Makki Sagheer, Ahmeed Suliman Farhanand Loay E. George	17	2013 International Journal of Computer Applications	Fast Intra-frame Compression for Video Conferencing using Adaptive Shift Coding	The adaptive shift coding technique reduces the number of bits which required implementing the data depending on the value.	Intra frame compression
6	Ashok nayak.B, Dr.E.Nagabhooshnam,	18	2014 IEEE	A Study of Efficient Block Matching Algorithms For Real-Time Video Compression Applications.	Comparison of block matching algorithm	Implementation on MPEG-2 Platform
7	Stamos Katsigiannis, Georgios Papaioannou, and Dimitris Maroulis	19	2015 arxiv Journal	CVC: The Contourlet Video Compression algorithm for real-time applications	By Using contourlet transformation and Graphics , Processing Units and CT coefficients reduction	Real-time compression can be achieved
8	Niras C. Vayalil, Joshua Haddrill and Yinan Kong	20	2016 IEEE	An Efficient ASIC Design of Variable-Length Discrete Cosine Transform for HEVC	Variable length DCT architecture for encoding video according to the HEVC/H.265 specifications.	Emphasis on hardware savings
9	Huaying Xue, Yuan Zhang, Yunong Wei	21	2016 IEEE	Fast ROI-Based HEVC Coding for Surveillance Videos	Region of Interest based HEVC coding	Usage in Surveillance Videos
10	S. Aparna, M. Ekambaram Naidu	22	2016 IEEE	Spatial Compression and Reconstruction of Digital Video Stream Using Morphological Filters	universal coder decoder procedure Sub sampling of video frames and their morphological processing are used for video Compression, transmission and reconstruction.	More Useful in still images
11	Vivek Diliprao Indrale, Mrs. Vidya N. More	23	2016 IEEE	Study of x265 and Genetic Motion Search Algorithm	Genetic Motion Search method is used instead of full search Algorithm (FSA) so bit rate is increased	The search points in the Genetic Motion Search algorithm are random.

12	Maher Jridi, and Pramod Kumar Meher	24	2017 IEEE	A Scalable Approximate DCT Architectures for Efficient HEVC Compliant Video Coding	4-point integer 1-D DCT has been used in HEVC coding	Degradation of video quality and slight increase in bit-rate
13	Zhao Wang, Shiqi Wang+, Jian Zhang, Shanshe Wang, Siwei Ma	25	2017 IEEE	Effective Quadtree Plus Binary Tree Block Partition Decision for Future Video Coding	Effective QTBT partition decision algorithm to achieve a good tradeoff between computational complexity and coding performance	It can reduce the coding complexity
14	Nijad A-Najdawi	26	2017 IEEE	Fast Block Matching Criterion for Real-Time Video Communication	A motion estimation algorithm in the frequency domain, where the new block matching method examines the similarities between a subset of frequencies in corresponding blocks.	Focuses on Motion estimation
15	Krishna Reddy Konda, Yonas Teodros Tefera, Nicola Conci, and Francesco G.B. De Natale	27	2017 IEEE	Real-time moving object detection and segmentation in H.264 video streams	moving object detection and segmentation, operating on H.264 bit streams.	Can be Implemented in H.264 stream
16	Liang Zhao, Zhihai He, *Fellow, IEEE*, Wenming Cao, and Debin Zhao, *Member, IEEE*	28	2018 IEEE	Real-Time Moving Object Segmentation and Classification from HEVC Compressed Surveillance Video	HEVC compressed domain moving object segmentation and classification	Usage in Surveillance Video

Differences in streaming and real-time video

The main difference between streaming video and real-time video conferencing is noticed when we start watching a streamed video segment (live or recorded). It doesn't start immediately. The reason is of course that video is being buffered to facilitate good quality. This buffering delay, that is typically tens of seconds long, allows for handling many of the major challenges of video over IP. Naturally, by having a buffer there will be time for retransmission of lost packets, something that can rarely be done in a video conferencing scenario. The long buffer also ensures that the network jitter (variations in transport time for IP packets) will not cause any quality issues. These two effects of buffering are well known by most people in the industry. However, there are some other less known benefits of high buffer latency that can be very helpful in
terms of delivering high quality media.

One of the major issues in video transmission is to manage the bit rate of the media to utilize the bandwidth available. If too high bit rate is used, there will be freezes in the picture and delay will keep building up even more. If, on the other hand, a lesser bit rate than the available bandwidth is used the best possible quality will not be delivered. The long buffers in streaming makes it easier to adapt to the available bandwidth since there is more room for error and adaptations don't need to be as quick as for real time video conferencing. An additional problem with the two-way nature of a video conference is that typically both sides will be limited by its uplink speed, which typically is an order of magnitude lower than the downlink speed. In streaming scenarios. There is only video going in the downlink direction.

Another important factor of video quality is that the audio and video are synchronized in time. This is also a much easier problem to solve if significant buffer delay is available to adjust the timing for the Audio or video stream the effect of long delay on the bit rate of the video (and audio) compression. Since audio and video are data sources that both are highly correlated between adjacent points in time, more efficient compression can be achieved if a longer portion of the signal is considered at each coding instance. For real time communication coding is typically happening every 20 ms to 60 ms while in streaming it is possible to consider up to several seconds of signal at each coding instance. The corresponding gain in compression efficiency is very significant and results in better quality for the same available bandwidth

Real time moving object moving object segmentation

Key purpose of video segmentation is to enable content-based representation by extracting objects of interest from a series of consecutive video frames. Mainly, it is required for high-level image understanding and scene explanation such as spotting and tracking of special events in surveillance video. For example, pedestrian and highway traffic can be regularized using density evaluations obtained by segmenting people and vehicles. By object segmentation, speeding and suspicious moving cars, road obstacles, strange activities can be detected. Tracking and recognition the object in a video can be done easily with the help of the segmentation methods such us background subtraction, temporal segmentation, edge detection, spatial segmentation and optical flow

V. MOTIVATION

Potentially everything. Each service will have to decide if it will send data using HEVC/H.265 instead of MPEG-4/H.264 or MPEG-2. Netflix has already begun using HEVC to stream 4K content to the few 4K TVs equipped to decode HEVC. New televisions coming this year from Sony and Vizio will support HEVC and will be able to stream 4K video from Netflix. Though they haven't mentioned Netflix deals . Individual cable companies will have to decide when to change both the data they are sending and the receiver to decode it to HEVC. But that likely won't happen for several years, if and when they broadcast in 4K. Though they aren't broadcasting in 4K. Most current cell phones and tablets don't have hardware compatible with HEVC, but some new devices are starting to support it. But that doesn't mean all mobile devices will eventually support HEVC. For instance, Google has announced its own 4K codec, called VP9 that will be used with YouTube. **More** compression is easy while **better** compression requires more thought and/or better technology.

VI. Conclusion AND FUTURE SCOPE

In real time video compression H.265/HEVC can become the universal standard codec if the supporting hardware is available. HEVC requires larger prediction units, costly motion estimation causes computationally expensive compared to its prior standards. Currently available video compression standard are also too complex so researchers can still explore the possibility of improvement in the area of video compression .By above critical review of the video compression and quality of video enhancement by fuzzy logic technique provides a room for researchers for better compression ratio.

In this review paper Real time video compression and use of fuzzy techniques to enhance the quality of video provides a scope for existing technologies this review work can be extended to convert it into a new approach using fuzzy logic in the arena of real time video compression.

ACKNOWLEDGMENT

I express my sincere gratitude towards Assistant Prof. Dr. Navin Prakash , Department of Computer Science and Engineering IFTM University , for his valuable suggestions and thanks to Head of Department of Computer Science and Engineering IFTM University, Moradabad for gave us necessary facilities.

REFERENCES

[1] Danny H. K. Tsang, Brahim Bensaou, and Shirley T. C. Lam, "Fuzzy-Based Rate Control for Real-Time MPEG Video", IEEE Transactions on Fuzzy Systems Vol. 6, Issue 4, pp. 504-516, 1998.

[2] Y.-S. Saw P.M. Grant J.M.Hannah, "Quality-optimized MPEG2 video data rate control using fuzzy logic techniques", IEE Proceedings - Vision, Image and Signal Processing Vol.145, Issue 3, pp. 179-186, 1998.

[3] Yun-Teng Roan and Pei-Yin Chen, "A Fuzzy Search Algorithm for the Estimation of Motion Vectors", IEEE Transactions on Broadcasting Vol. 46, Issue 2 , pp. 121-187, 2000.

[4] Ming-Chieh Chi, Mei-Juan Chen and Ching-Ting Hsu, "Region-of-interest video coding by fuzzy control for h.263+ standard", In the Proceedings of the IEEE Conferences 2004 IEEE International Symposium on Circuits and Systems , Vancouver, BC, Canada, pp. 93 - 96, 2004.

[5] Pei-Jun Lee and Ming-Long Lin, " Fuzzy Logic Based Temporal Error Concealment for H.264 Video ", ETRI Journal Vol. 28, Number 5, pp. 574-582, 2006.

[6] Mehdi Rezaei, Miska M. Hannuksela, and Moncef Gabbouj, , " Semi-Fuzzy Rate Controller forVariable Bit Rate Video", IEEE Transactions on Circuits and Systems for Video Technology Vol.18, Issue 5, pp. 633-65, 2008.

[7] WU Jing, DU Xin, ZHU Yun-fang, GU Wei-kang, "Adaptive Fuzzy Filter Algorithm for Real-time Video Denoising", In the Proceedings of the IEEE Conferences 2008 9th International Conference on Signal Processing , Beijing, China , pp. 1287 - 1291, 2008.

[8] S.M.R. Soroushmehr S. Samavi M. Saraee, "Fuzzy block matching motion estimation for video compression", In the Proceedings of the IEEE Conferences 2009 IEEE 9th Malaysia International Conference on Communications (MICC) , Kuala Lumpur, Malaysia, pp. 1287 - 1291, 2008.

[9] Junhua Chai, Jun Ying and Li Li, "A Fuzzy Video Pre-filtering Method for Impulse Noise Reduction", In the Proceedings of the IEEE Conferences 2009 International Conference on Test and Measurement , Hong Kong, China, pp. 176 - 183, 2009.

[10] C. Solana-Cipres, L. Rodriguez-Benitez1 J. Moreno-Garcia2 L. Jimenez1 G. Fernandez-Escribano, " Real-time segmentation of moving objects in H.264 compressed domain with dynamic design of fuzzy sets", IFSA-EUSFLAT Journal , pp. 19-24, 2009.

[11] Zhan Xuefeng Zhu Xiuchang, " A novel temporal error concealment method based on Fuzzy reasoning for H.264 ", Vol.27 No.2 JOURNAL OF ELECTRONICS (CHINA), pp. 197-205, 2010.

[12] Suvojit Acharjee Sheli Sinha Chaudhuri, " Fuzzy Logic Based Four Step Search Algorithm for Motion Vector Estimation ", I.J. Image, Graphics and Signal Processing, Vol. 4, pp. 49-55, 2012.
[13] T. Fryza, "Introduction to Implementation of Real Time Video Compression Method", In the Proceedings of the IEEE Conferences 2008 15th International Conference on Systems, Signals and Image Processing , Bratislava, Slovakia, pp. 217 - 220, 2008.
[14] J. C. Galan-Hernandez, V. Alarcon-Aquino, O. Starostenko, J. M. Ramirez-Cortes "Wavelet-Based Foveated Compression Algorithm for Real-Time Video Processing", In the Proceedings of the IEEE Conferences 2010 Electronics, Robotics and Automotive Mechanics Conference ,Morelos, Mexico, pp. 405 - 410, 2010.
[15] Gary J. Sullivan, Jens-Rainer Ohm, Woo-Jin Han and Thomas Wiegand , "Overview of the High Efficiency Video Coding (HEVC) Standard" IEEE Transactions on Circuits and Systems for Video Technology Vol. 22, Issue: 12, pp. 1649 – 1668, 2012
[16] Stamos Katsigiannis, Dimitris Maroulis, Georgios , Papaioannou , "A GPU based real-time video compression method for video conferencing", In the Proceedings of the IEEE Conferences 2013 18th International Conference on Digital Signal Processing (DSP) , Fira, Greece, pp. 1 - 6, 2013.
[17] Ali Makki Sagheer, Ahmeed Suliman Farhanand Loay E. George, "Fast Intra-frame Compression for Video Conferencing using Adaptive Shift Coding" International Journal of Computer Applications , Vol. 81, No.8, pp. 29 – 33, 2013
[18] Ashok nayak. B , E. Nagabhooshnam "A study of efficient Block Matching Algorithms for real-time video compression applications", In the Proceedings of the IEEE Conferences 2014 International Conference on Electronics and Communication Systems (ICECS) , Coimbatore, India , pp. 1 - 5, 2014.
[19] Stamos Katsigiannis, Georgios Papaioannou, and Dimitris Maroulis, "CVC: The Contourlet Video Compressionalgorithm for real- time applications" arxiv Journal, pp. 1 – 22, 2015
[20] Niras C. Vayalil, Joshua Haddrill and Yinan Kong, "An Efficient ASIC Design of Variable-Length Discrete Cosine Transform for HEVC", In the Proceedings of the IEEE Conferences 2016 European Modelling Symposium (EMS) , Pisa, Italy , pp. 229 - 233, 2016.
[21] Huaying Xue, Yuan Zhang, Yunong Wei, " Fast ROI-Based HEVC Coding for Surveillance Videos", In the Proceedings of the IEEE Conferences 2016 19th International Symposium on Wireless Personal Multimedia Communications (WPMC), Shenzhen, China, pp. 299 - 304, 2016.
[22] S. Aparna, M. Ekambaram Naidu, "Spatial Compression and Reconstruction of Digital Video Stream Using Morphological Filters", In the Proceedings of the IEEE Conferences 2016 2nd International Conference on Next Generation Computing Technologies (NGCT) , Dehradun, India, pp. 777 - 781, 2016.
[23] Vivek Diliprao Indrale, Mrs. Vidya N. More, "Study of x265 and Genetic Motion Search Algorithm", In the Proceedings of the IEEE Conferences 2016 International Conference on Computing Communication Control and automation (ICCUBEA) , Pune, India, pp. 1 - 5, 2016.
[24] Maher Jridi, and Pramod Kumar Meher, "A Scalable Approximate DCT Architectures forEfficient HEVC Compliant Video Coding", IEEE Transactions on Circuits and Systems for Video Technology Vol. 27, Issue 8, Aug. 2017 pp. 1815 – 1825, 2017
[25] Zhao Wang, Shiqi Wang, Jian Zhang, Shanshe Wang, Siwei Ma, "Effective Quadtree Plus Binary Tree Block Partition Decision for Future Video Coding", In the Proceedings of the IEEE Conferences 2017 Data Compression Conference (DCC) , Snowbird, UT, USA, pp. 23 - 32, 2017.
[26] Nijad A-Najdawi, "Fast Block Matching Criterion for Real-Time VideoCommunication", In the Proceedings of the IEEE Conferences 2017 International Conference on New Trends in Computing Sciences (ICTCS) , Amman, Jordan, pp. 327 - 332, 2017.
[27] Krishna Reddy Konda, Yonas Teodros Tefera, Nicola Conci, and Francesco G.B. De Natale, "Real-time moving object detection and segmentation in H.264 video streams", In the Proceedings of the IEEE Conferences 2017 IEEE International Symposium on Broadband Multimedia Systems and Broadcasting (BMSB) , Cagliari, Italy, pp. 1 - 6, 2017.
[28] Liang Zhao, Zhihai He, Wenming Cao, and Debin Zhao, "Real-Time Moving Object Segmentation and Classification from HEVC Compressed Surveillance Video", IEEE Transactions on Circuits and Systems for Video Technology Vol. 28, Issue 6, , pp. 1346-1357, 2018.

Authors Profile

Upendra Kumar Srivastava is prsung Ph.D. from IFTM University. He received his M.Tech (CSE) degree from Dr. A.P.J. Kalam Technical University U.P. in 2016 his area of interest is Image Processing and Soft Computing particularly Fuzzy Logic.

Navin Prakash is working as an Assistant Professor in Department of Computer science and Engineering, IFTM, University Moradabad. His current research interests include Image Processing, Machine learning.

