CHAPTER -2

LITERATURE REVIEW

2.1 Introduction

Compression is a reversible conversion (encoding) of data that contains fewer bits. This allows a more efficient storage and transmission of the data. The inverse process is called decompression (decoding). The Compression includes software and hardware that can encode and decode. Both combined form a codec and should not be confused with the terms data container or compression algorithms.

Lossless compression allows a 100% recovery of the original data. It is usually used for text or executable files, where a loss of information is a major damage. These compression algorithms often use statistical information to reduce redundancies. Huffman-Coding and Run Length Encoding are two popular examples allowing high compression ratios depending on the data.

Using lossy compression does not allow an exact recovery of the original data. Nevertheless it can be used for data, which is not very sensitive to losses and which contains a lot of redundancies, such as images, video or sound. Lossy compression allows higher compression ratios than lossless compression.

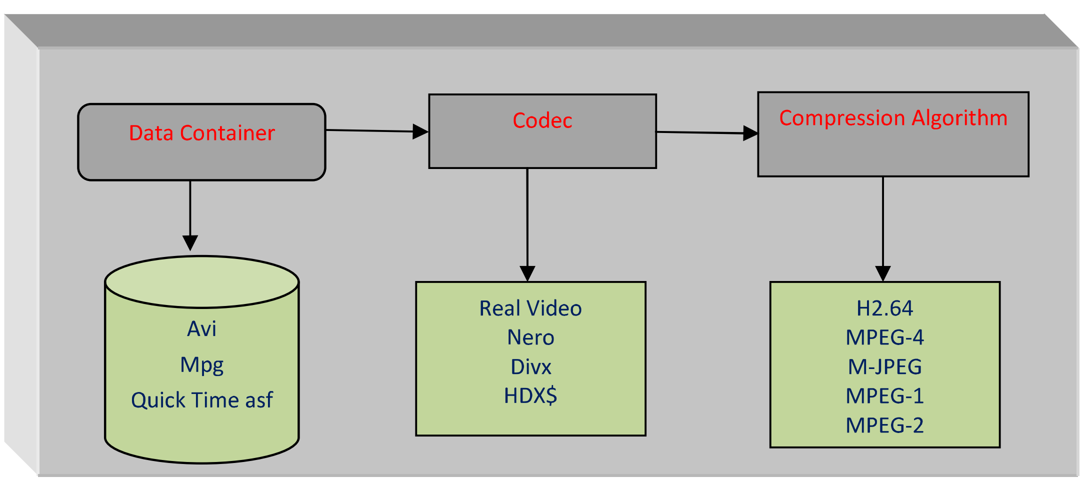

Fig 2.1. Image Compression Formats

2.1.1 Need of Compression

A simple calculation shows that an uncompressed video produces an enormous amount of data: a resolution of 720x576 pixels (PAL), with a refresh rate of 25 fps and 8-bit colour depth, would require the following bandwidth:

720 x 576 x 25 x 8 + 2 x (360 x 576 x 25 x 8) = 1.66 Mb/s (luminance + chrominance)

For High Definition Television (HDTV):

1920 x 1080 x 60 x 8 + 2 x (960 x 1080 x 60 x 8) = 1.99 Gb/s

Even with powerful computer systems (storage, processor power, network bandwidth), such data amount cause extreme high computational demands for managing the data. Fortunately, digital video contains a great deal of redundancy. Thus it is suitable for compression, which can reduce these problems significantly. Especially lossy compression techniques deliver high compression ratios for video data. However, one must keep in mind that there is always a trade-off between data size (therefore computational time) and quality. The higher the compression ratio, the lower the size and the lower the quality. The encoding and decoding process itself also needs computational resources, which have to be taken into consideration. It makes no sense, for example for a real-time application with low bandwidth requirements, to compress the video with a computational expensive algorithm which takes too long to encode and decode the data.

2.2 Standardization Organizations

There are two important organizations that develop image and video compression standards: International Telecommunications Union (ITU) and International Organization for Standardization (ISO). Formally, ITU is not a standardization organization. ITU releases its documents as recommendations, for example “ITU-R Recommendation BT.601” for digital video. ISO is a formal standardization organization, and it further cooperates with International Electro Technical Commission (IEC) for standards within areas such as IT. The latter organizations are often referred to as a single body using “ISO/IEC”.

The fundamental difference is that ITU stems from the telecommunications world, and has chiefly dealt with standards relating to telecommunications whereas ISO is a general standardization organization and IEC is a standardization organization dealing with electronic and electrical standards. Lately however, following the ongoing convergence of communications and media and with terms such as “triple play” being used (meaning Internet, television and telephone services over the same connection), the organizations, and their members – one of which is Axis Communications – have experienced increasing overlap in their standardization efforts.

2.3 Basics of Compression

Compression basically means reducing image data. As mentioned previously, a digitized analog video sequence can comprise of up to 165 Mbps of data. To reduce the media overheads for distributing these sequences, the following techniques are commonly employed to achieve desirable reductions in image data:

- Reduce color nuances within the image
- Reduce the color resolution with respect to the prevailing light intensity
- Remove small, invisible parts, of the picture
- Compare adjacent images and remove details that are unchanged between two images

The first three are image based compression techniques, where only one frame is evaluated and compressed at a time. The last one is a video compression technique where different adjacent frames are compared as a way to further reduce the image data. All of these techniques are based on an accurate understanding of how the human brain and eyes work together to form a complex visual system.

As a result of these subtle reductions, a significant reduction in the resultant file size for the image sequences is achievable with little or no adverse effect in their visual quality. The extent, to which these image modifications are humanly visible, is typically dependent upon the degree to which the chosen compression technique is used. Often 50% to 90% compression can be achieved with no visible difference, and in some scenarios even beyond 95%.

2.4 Latency

Compression involves one or several mathematical algorithms that remove image data. When the video is to be viewed other algorithms are applied to interpret the data and view it on the monitor. Those steps will take a certain amount of time. That delay is called compression latency. The more advanced compression algorithm, the higher the latency. When using video compression and several adjacent frames are being compared in the compression algorithm, more latency is introduced. For some applications, like compression of studio movies, compression latency is irrelevant since the video is not watched live. In surveillance and security using live monitoring, especially when PTZ and dome cameras are being used, low latency is essential.

2.5 Basic steps in Digital Image Compression

2.5.1 Block Transform

The first step in image compression is to divide the image into small blocks, usually of size 8x8 pixels. Then, Discrete Cosine Transform (DCT) is applied on each block to convert each pixel value into frequency domain. It takes 64 input values and yields 64 frequency domain coefficients. This transform is fully reversible; the original block can be reconstructed by applying an InverseDCT(IDCT).

DCT not only converts pixels into frequencies but also does it in such a way that lower frequencies appear at the top-left side of the block, while higher frequencies appear at the bottom right. Human eye is sensitive to only low frequencies and hence the future steps tend to discard the high frequency values to achieve compression. Hence, DCT helps separate more perceptible information from less perceptible information.

DCT and IDCT are computationally intensive operations, and can take about 30% of processor cycles. But, their memory requirement is very small as they operate on small blocks. Hence, it is ideal to implement them in dedicated hardware co-processors.

2.5.2 Quantization

Quantization is used to discard perceptibly insignificant information. It basically converts each real DCT coefficient to an integer by scaling it by a factor and then discarding the digits after the decimal point. For each coefficient, a scaling factor is chosen in such a way that there is no perceptible change even after discarding digits after the decimal point.

Note that the inverse quantization cannot recover the original value completely. Quantization and inverse quantization can take anywhere between 3% and 15% of processor cycles and have only small memory requirement.

2.5.3 Coding

The next step in the compression process is to encode the DCT coefficients using as few bits as possible. After quantization, most of the DCT coefficients are zeros. This is true for most high frequency DCT coefficients. To take advantage of this, consecutive zeros are grouped and the number of zeros in the group is encoded. This process is called "Run-Length encoding”. To facilitate run-length encoding, the DCT coefficients are encoded in a zig-zag fashion, starting from top-left and ending at bottom-right.

By doing this, most of the high-frequency coefficients at the bottom-right that usually are zeros can be encoded using a single number. Run-length encoding is followed by “Variable-Length Coding”. In VLC, more frequently occurring symbols (either a run-length encoded number or a DCT-coefficient) are coded using fewer bits, and infrequent symbols are encoded using more bits. “Huffman coding” is an example of variable-length coding. The bits to be used for encoding a symbols, is found from a look-up table that gives code-words for each symbol and the length.

Variable-length decoding is a much more computationally intensive operation. The simple VLD implementation requires a table lookup and some decision making for each input bit, which finally evaluates to about 11 operations per bit. So, the processing requirement of VLD is proportional to the bit-rate of the video-compression bit-rate. Also, since a lookup table is used it is memory intensive as some lookup tables can take several kilo-bytes of memory. One disadvantage of VLC, is that an error of even one-bit in the bit-stream corrupts the entire stream. To circumvent this, resynchronization markers are used. Whenever the decoder finds a bit-error, it fast-forwards till a resynchronization marker and then again starts decoding.

2.6 Video Quality Measure

In order to evaluate the performance of video compression coding, it is necessary to define a measure to compare the original video and the video after compressed. Most video compression systems are designed to minimize the mean square error (MSE) between two video sequences $\Psi 1$ and $\Psi 2$, which is defined as

$$\text{MSE}=\sigma_e^2 = \frac{1}{N}\sum_t \sum_{x,y} \quad [\varphi_1 1\ (x,y,t) - \varphi_2(x,y,t)]^2$$

where N is the total number of frames in either video sequences. Instead of the MSE, the peak-signal-to-noise ratio (PSNR) in decibel (dB) is more often used as a quality measure in video coding, which is defined as

$$\text{PSNR} = 20 \log 10 \frac{255}{MSE}$$

it is worth noting that one should compute the MSE between corresponding frames, average the resulting MSE values over all frames, and finally convert the MSE value to PSNR.

2.7 Implementation Strategies

There are a number of techniques to implement the video compression and these techniques can be broadly divided into two categories: hardware-based implementation and software-based implementation.

2.7.1 Hardware-Based Approach

The most common approach is to design a dedicated VLSI circuit for video compression. One can have function specific hardware, such as associated inverse operations, DCT, VLC and block matching. Due to the exploitations of the data flow of the algorithm and special control, the processing capability of these approaches can be increased tenfold compared to those of conventional microprocessors. However, function specific approaches provide limited flexibility and cannot be modified for further developments. In addition, the architecture design usually requires a regular control paradigm and data flow of the algorithms that may not be useful for solving the complexities of circuit designs. Furthermore, the complexity limitations of the circuit design, such as processing speed, throughput, the number of translators, silicon area, also restrict its implementation potential for growing multimedia applications. A more cost-effective alternative is provided by programmable processors, such as programmable DSP or VSP. Such an approach can execute different tasks under software control, it can avoid cost intensive hardware redesign. Programmable processors are flexible in the way it allows the implementation of various video compression algorithms without the need for a hardware redesign. Moreover, multiple algorithms can be executed on the same hardware and their performance can be optimized as well. Consequently, their implementation time and cost increase accordingly. Furthermore, they also incur significant costs in software development and system integration. Usually, programmable processors require silicon area for program storage and control unit, and dissipate more power than dedicated VLSI solutions.

2.7.2 Software-Based Approach

Software-based approaches are becoming more popular because the performance of general-purpose processors has been increasing rapidly. Further, more and more emerging multimedia standards emphasize high-level interactivity, extensibility, and flexibility, posing significant opportunities for software-based solutions Furthermore, the rapid evolution of multimedia techniques has dramatically shortened the required time for market making it very difficult to come up with a new hardware design for each updated technique. The inherent modular nature of various video compression algorithms allows experimenting and hence improving various parts of the encoder independently, including ME, DCT algorithm and rate-controlled coding. The major advantage of using the software-based approach is that it allows incorporating new research ideas and algorithms in the encoding process for achieving a better picture quality at a reduce bit rate for a desired level of picture quality, or on given bit rate. The software-based approach is also flexible in that it allows tuning of various parameters for multiple passes for optimization. Some more benefits of software-based approach are flexibility to adapt to the continuing changes in multimedia applications and portability. Encoding is more challenging due to enormous amount of computation required, whereas decoding can be done easily in software. Real-time performance for high-quality profiles is still quite difficult, but encoding for simple video profiles of various standards can now be done on a single processor. To speed up the compression, a natural alternative is used to utilize the accumulated processing capability of parallel processing. However, parallelism can be exploited in different ways, there is no unique philosophy for the best solution, ranging from simultaneous instructions execution within massively parallel processors (MPPs) and a single processor, to distributed networks. It is important to recognize that parallel processing alone may not be enough in software-based implementation, this includes efficient algorithms for DCT, a fast ME and other parts of the encoder . In addition, low-level programming primitives that take advantage of the machine architecture must be harnessed to accelerate the computation. Finally, several issues should be addressed in software-based parallel processing such as I/O, memory access, and achieving better rate control.

2.8 Issue on Emerging Technologies

The predicted growth in demand for bandwidth, driven largely by video applications, is probably greater now than it has ever been. There are four primary drivers for this:

1. Recently introduced formats such as 3-D and multiview, coupled with pressures for increased dynamic range, spatial resolution and framerate, all require increased bit-rate to deliver improved levels of immersion or interactivity.

2. Video-based web traffic continues to grow and dominate the internet through social networking and catch up TV. In recent years, Youtube has accounted for 27% of all video

traffic and, by 2015, it is predicted that there will be 700 billion minutes of video downloaded. That represents a full-length movie for every person on the planet.

3. User expectations continue to drive flexibility and quality, with a move from linear to nonlinear delivery. Users are demanding My-Time rather than Prime-Time viewing.

4. Finally new services, in particular mobile delivery through 4G/LTE to smart phones. Some mobile network operators are predicting the demand for bandwidth to double every year for the next 10 years!

2.9 H.264 Features

H.264/AVC/MPEG-4 Part 10 contains a number of new features that allow it to compress video much more effectively than older standards and to provide more flexibility for application to a wide variety of network environments. In particular, some such key features include:

- Multi-picture inter-picture prediction.
- Variable block-size motion compensation.
- The ability to use multiple motion vectors per macro block.
- Quarter-pixel precision for motion compensation.
- Spatial prediction from the edges of neighbouring blocks for "intra" coding.
- Flexible interlaced-scan video coding features.
- New transformation design features.
- An entropy coding design including context adaptive binary arithmetic coding and context adaptive variable-length coding.

2.10 Related and Previous Work

According to Missu Toshie, et al. [MT+2013], they have developed a prototype real-time

UHDTV video coding system based on our proposed reconstructive video coding paradigm. In the paradigm, an image reduction process before a conventional lossy encoder suppresses non-linear distortions such as blocking artifacts alleviating the compression ratio at the encoder. On the receiver side, the super-resolution based reconstruction process recovers the resolution of the decoded image by restoring and/or substituting the folded/omitted over-Nyquist components in the reduction. The paradigm is capable of transmitting side data that optimally controls the reconstruction process by trying the reconstruction in advance on the transmission side. They have focused on the hardware implementation of the reduction and super-resolution processes on FPGA-based 4K video processing PCI Express cards. They achieved real-time high-quality transmission of UHDTV video at a compression ratio of around 500:1.

Based on Lee, yu-Hsuan, et al. [LY+2010] Modern video coding system, the memory bandwidth of external memory becomes more and more important and almost dominates the system performance. To save the memory bandwidth, the embedded compression (EC) technique is integrated into video coding system. In their paper, the bandwidth-efficient EC algorithm is proposed. It comprises three core techniques: side-oriented prediction, adjusted binary code (ABC) entropy coding, and two-level rate control scheme. The side-oriented prediction can save the extra bits for the representation of prediction residual. The prediction residual is allocated to an efficient codeword by ABC entropy coding. With two-level rate control scheme, not only the compression ratio (CR) can be precisely controlled, but also the visual quality is maintained. Experiment results show this work achieves the PSNR drop of 1.33%, and the error between CR and Target CR(TCR) is as minor as 1.60%. Consequently, this work is quite suitable for saving memory bandwidth in video coding system.

According to Nair and Benny[RB+2010] Video Surveillance systems are becoming increasingly popular due to the emergence of high-speed wireless Internet (such as WiMax and LTE), band width efficient video compression schemes (such as H.264), and low-cost (and high-resolution) IP video cameras. They present two applications of an advanced surveillance system, specifically in suspicious activity detection and human fall detection, for both indoor and outdoor environments. The implemented prototype captures and analyzes live high-definition (HD) video that is streamed from a remote camera. They have shown that by combining the strengths of ellipse modelling, shadow removal and other novel algorithms, the false alarms in the detection can be significantly reduced.

In view of Shao. F, Yu. M., et al. [SY+2012]. Three-dimensional (3D) video technologies are becoming increasingly popular, as they can provide high quality and immersive experience to end users, where depth maps are employed to generate the virtual views by depth-image-based rendering technique. However, how to reduce the compression and rendering complexities for depth maps while maintaining high rendering quality is still unresolved. In their study, a novel depth map compression and depth-aided view rendering method is proposed. In their proposed method, depth maps are represented with different layers and compressed with different macro block-mode decision procedure, and several optimisation techniques, including spatio-temporal consistent warping, colour correction and temporal consistent hole filling are embedded into the view rendering framework. Experimental results show that compared with the traditional method, the proposed method can reduce more than 79% compression computational complexity and more than 45% rendering computational complexity, while maintaining high rendering quality.

The authors Shehata, Khaled, et al. [SK+2011] say, High definition videos contain huge data to be transmitted and received, which is difficult to be sent and stored. In this paper a compression technique based on binary motion vector technique is used to compress HD videos. In which the process of searching for the best matching block for each current block occurs. The proposed technique keeps HD quality with at least Peak Signal to Noise Ratio 62 dB and along with 26% compression ratio on gray scale. The proposed technique is implemented on a Xilinx Vertex 2 2V250fg456 FPGA. The maximum operating speed of the hardware is 63.8 MHz. The FPGA utilization is 12.37% of total CLB slices and 8.61% of total latches.

According to Misu, Toshie, et al. [MT+2013], they have developed a prototype real-time UHDTV video coding system based on their proposed reconstructive video coding paradigm. In the paradigm, an image reduction process before a conventional lossy encoder suppresses non-linear distortions such as blocking artifacts alleviating the compression ratio at the encoder. On the receiver side, the super-resolution based reconstruction process recovers the resolution of the decoded image by restoring and/or substituting the added/omitted over-Nyquist components in the reduction. The paradigm is capable of transmitting side data that optimally controls the reconstruction process by trying the reconstruction in advance on the transmission side. This paper deals with the hardware implementation of the reduction and super-resolution processes on FPGA-based 4K video processing PCI Express cards. They have achieved real-time high-quality transmission of UHDTV video at a compression ratio of around 500:1.

According to Nguyen, Viet-Anh, et al. [NV+2013], efficient techniques to compress a depth video by taking into account coding artifacts, spatial resolution, and dynamic range of the depth data is proposed. Due to abrupt signal changes on object boundaries, a depth video compressed by conventional video coding standards often introduces serious coding artifacts over object boundaries, which severely affect the quality of a synthesized view. We suppress the coding artifacts by proposing an efficient post processing method based on a weighted mode filtering and utilizing it as an in-loop filter. In addition, their proposed filter is also tailored to efficiently reconstruct the depth video from the reduced spatial resolution and the low dynamic range. The down/up sampling coding approaches for the spatial resolution and the dynamic range are used together with the proposed filter in order to further reduce the bit rate. They had verified the proposed techniques by applying them to an efficient compression of multiview-plus-depth data, which has emerged as an efficient data representation for 3-D video. Experimental results show that their proposed techniques significantly reduce the bit rate while achieving a better quality of the synthesized view in terms of both objective and subjective measures.

According to Gao, Jun-Wei, et al. [GJ+2009], Embedded video surveillance has become a large market as the number of installed cameras around us can show. At the same time, video compression technique has been rapidly developed in recent years. As a new generation of video coding compression standard, extensive attention has been given to H.264/AVC. On the other hand, because of good cost effective and low-power consumption, DSP is widely used in the fields of embedded system and multimedia processing. There are needs for embedded video surveillance that can make use of the video compression technique in more efficiency security systems. In their paper, a scheme of design and implementation about video surveillance system based on H.264/AVC is designed and described. It is built on ADSP-BF561, with a wide range of peripheral interface, to achieve several functions, such as video compression, data stream control, network transmission, PTZ and so on. The experimental results indicate that the stability and other expected features of system, including Real-time surveillance video encoding, can be ensured

The authors Lai, Yeong-Kang, et al. [LY+2011] say that Variable Length Coding (VLC) is a widely used technique in digital video compression systems. It is known for its efficient compression, but is susceptible to noisy environments. Due to the increased demand for multimedia systems to be portable, power consumption and power saving become important issues. Current ITU H.263+ and ISO MPEG-4 standards have used reversible variable length coding (RVLC) which provides greater error robustness than non-reversible counterparts (VLC) due to the growing need for wireless exchange of compressed image and video signals over noisy channels. In their paper, a new method for RVLC decoding is described. Since the special structure of RVLC code words, the decoding techniques that are common for regular VLC are less efficient when used with RVLC. The new method uses simple logical operations to determine the length of code words quickly, and then code words are decoded. It is easily implemented with hardware. They propose a VLSI architecture based on this new method. The architecture also uses the technique of table partitioning. The experimental result shows that our architecture can achieve lower power consumption without sacrificing the quality of the performance. Their proposed architecture has been implemented using standard cell methodology for TSMC 0.18um 1P6M technology. The chip implementation results show that proposed architecture can work at 100MHz and its power consumption is only 46.69 uW/MHz.

In view of Akin, Abdulkadir, et al. [AA+2011] Motion Estimation (ME) is the most computationally intensive part of video compression and video enhancement systems.

Therefore, in their paper, they propose comparison prediction (CP) technique for reducing the power consumption of block matching (BM) ME hardware. CP technique reduces the power consumption of absolute difference operations performed by BM ME hardware. CP technique can easily be used in all BM ME hardware. In this paper, they have applied it to a 256 processing element fixed block size ME hardware implementing full search algorithm. It reduced the average dynamic power consumption of this ME hardware by 2.2% with no Peak Signal-to-Noise Ratio (PSNR) loss and by 9.3% with 0.04% PSNR loss on a XC2VP30-7 FPGA.

According to Rajasekhar. H, and B. Prabhakara Rao. [RP2012], Wavelet based compression is mostly utilized to compress the videos. Most of the wavelet-based compression techniques first determines the coefficients and then performs a threshold-based operation to obtain the compressed video. In such works, there is a lack of analysis in selecting an appropriate threshold value. To overcome such drawback presented in the existing methods, they have proposed a video compression technique with an adaptive threshold selection method to compress the videos in their paper. The simulation results show the effectiveness of proposed video compression technique. The performance of the video compression technique is evaluated by comparing the result of proposed technique with the existing video compression technique. The comparison result shows that the enhanced quality of the image as well as the compression ratio of the proposed technique.

According to Ponlatha S., and R. S. Sabeenian [PS2013], in order to ensure compatibility among video codecs from different manufacturers and applications and to simplify the development of new applications, intensive efforts have been undertaken in recent years to define digital video standards Over the past decades, digital video compression technologies have become an integral part of the way we create, communicate and consume visual information. Digital video communication can be found today in many application scenarios such as broadcast services over satellite and terrestrial channels, digital video storage, wires and wireless conversational services and etc. The data quantity is very large for the digital video and the memory of the storage devices and the bandwidth of the transmission channel are not infinite, so it is not practical for us to store the full digital video without processing. For instance, we have a 720 x 480 pixels per frame,30 frames per second, total 90 minutes full color video, then the full data quantity of this video is about 167.96 G bytes. Thus, several video compression standards, techniques and algorithms had been developed to reduce the data quantity and provide the acceptable quality as possible as can. Thus they often represent an optimal compromise between performance and complexity. Their paper describes the main features of video compression standards, discusses the emerging standards and presents some of its main characteristics.

In view of authors Mostafa, Atahar et al. [MA+2014],Video capsule endoscopy is a non-invasive technique to receive images of intestine for medical diagnostics. The main design challenges of endoscopy capsule are accruing and transmitting acceptable quality images by utilizing as less hardware and battery power as possible. In order to save wireless transmission power and bandwidth, an efficient image compression algorithm needs to be implemented inside the endoscopy electronic capsule. In their paper, an integer discrete cosine- transform (DCT) based algorithm is presented that works on a low-complexity color-space specially designed for wireless capsule endoscopy application. First of all, thousands of human endoscopic images and video frames have been analyzed to identify special intestinal features present in those frames. Then a color space, referred as YEF, is used. The YEF converter is lossless and takes only a few adders and shift operation to implement. A low-cost quantization scheme with variable chroma sub-sampling options is also implemented to achieve higher compression. Comparing with the existing works, their proposed transform coding based compressor performs strongly with an average compression ratio of 85% and a high image quality index, peak-signal-to-noise ratio (PSNR) of 52 dB.

In view of Chen, Yue, et al. [CY+2013], A novel filtering approach that naturally combines information from both intra-frame and motion compensated referencing for efficient prediction is proposed to fully exploit the spatio-temporal correlations of video signals, thereby achieving superior compression performance. Inspiration was drawn from their recent work on extrapolation filter based intra prediction, which views the spatial signal as a non-separable first-order Markov process and employs a 3-tap recursive filter to effectively capture the statistical characteristics. Their work significantly extends the scope to further incorporate motion compensated reference in a filtering framework, whose coefficients were optimized via a “k-modes”-like iteration that accounts for various factors in the compression process including variation in statistics in the prediction loop, to minimize the rate-distortion cost. Experiments validate the efficacy of the proposed spatio-temporal approach, which translates into consistent coding performance gains.

The authors Liu, Limin, et al. [LL+2009] say that, Video surveillance has been widely used in recent years to enhance public safety and privacy protection. A video surveillance system that deals with content analysis and activity monitoring needs efficient transmission and storage of the surveillance video data. Video compression techniques can be used to achieve this goal by reducing the size of the video with no or small quality loss. State-of-the-art video compression methods such as H.264/AVC often lead to high computational complexity at the encoder, which is generally implemented in a video camera in a surveillance system. This can significantly increase the cost of a surveillance system, especially when a mass deployment of end cameras is needed. In their paper, they have discussed the specific considerations for surveillance video compression. They present a surveillance video compression system with low complexity encoder based on Wyner–Ziv coding principles to address the trade off between computational complexity and coding efficiency. In addition, we propose a backward-channel aware Wyner–Ziv (BCAWZ) video coding approach to improve the coding efficiency while maintaining low complexity at the encoder. The experimental results show that for surveillance video contents, BCAWZ can achieve significantly higher coding efficiency than H.264/AVC INTRA coding as well as existing Wyner–Ziv video coding methods and is close to H.264/AVC INTER coding, while maintaining similar coding complexity with INTRA coding. This shows that the low motion characteristics of many surveillance video contents and the low-complexity encoding requirement make our scheme a particularly suitable candidate for surveillance video compression. They further propose an error resilience scheme for BCAWZ to address the concern of reliable transmission in the backward-channel, which is essential to the quality of video data for real-time and reliable object detection and event analysis.

According to Ahmad, Junaid, et al. [AJ+2009], Wireless video sensor networks are anticipated to be deployed to monitor remote geographical areas. To save energy in bit transmissions/receptions over a video sensor network, the captured video content needs to be encoded before its transmission to the base station. However, video encoding is an inherently complex operation that can cause a major energy drain at battery-constrained sensors. Thus a systematic evaluation of different video encoding options is required to allow a designer to choose the most energy-efficient compression technique for a given video sensing application scenario. In their paper, they empirically evaluate the energy efficiencies of predictive and distributed video coding paradigms for deployment on real-life sensor motes. For predictive video coding, our results show that despite its higher compression efficiency, inter video coding always depletes much more energy than intra coding. Therefore, they propose to use image compression based intra coding to improve energy efficiency in the predictive video coding paradigm. For distributed video coding, our results show that the Wyner-Ziv encoder has consistently better energy efficiency than the PRISM encoder. They propose minor modifications to PRISM and Wyner-Ziv encoders which significantly reduce the energy consumption of these encoders. For all the video encoding configurations evaluated in their paper, the results reveal the counter-intuitive and important finding that the major source of energy drain in WSNs is local computations performed for video compression and not video transmission.

The authors Ouni, Tarek, et al. [OT+2009] say that, generally, video signal has high temporal redundancies due to the high correlation between successive frames. Actually, this redundancy has not been exploited enough by current video compression techniques. In their paper, they present a new video compression approach which tends to hard exploit the pertinent temporal redundancy in the video frames to improve compression efficiency with minimum processing complexity. It consists on a 3D to 2D transformation of the video frames that allows exploring the temporal redundancy of the video using 2D transforms and avoiding the computationally demanding motion compensation step. This transformation turns the spatial temporal correlation of the video into high spatial correlation. Indeed, this technique transforms each group of pictures to one picture eventually with high spatial correlation. Thus, the decorrelation of the resulting pictures by the DCT makes efficient energy compaction, and therefore produces a high video compression ratio. Many experimental tests had been conducted to prove the method efficiency especially in high bit rate and with slow motion video. Their proposed method seems to be well suitable for video surveillance applications and for embedded video compression systems.

According to Suresh G., P. Epsiba, et al. [SE+2010], in their paper, they propose a low complex Scalable ACC-DCT based video compression approach which tends to hard exploit the pertinent temporal redundancy in the video frames to improve compression efficiency with less processing complexity. Generally, video signal has high temporal redundancies due to the high correlation between successive frames. Actually, this redundancy has not been exposed enough by current video compression techniques. Their model consists on 3D to 2D transformation of the video frames that allows exploring the temporal redundancy of the video using 2D transforms and avoiding the computationally demanding motion compensation step. This transformation turns the spatial temporal correlation of the video into high spatial correlation. Indeed, this technique transforms each group of pictures (GOP) to one picture (Accordion Representation) eventually with high spatial correlation. Their model is also incorporated with up/down sampling method (SVC) which is based on a combination of the forward and backward type discrete cosine transform (DCT) coefficients. As this kernel has various symmetries for efficient computation, a fast algorithm of DCT-based Scalability concept is also proposed. For further improvement of the scalable performance, an adaptive filtering method is introduced, which applies different weighting parameters to DCT coefficients. Thus, the decorrelation of the resulting pictures by the DCT makes efficient energy compaction, and therefore produces a high video compression ratio. Many experimental tests had been conducted to prove the method efficiency especially in high bit rate and with slow motion video. Their proposed method seems to be well suitable for video surveillance applications and for embedded video compression systems.

The authors Zakariya S. M., and M. Inamullah [ZI 2012],say, The digital video application has become increasingly popular in mobile terminals such as cellular phones and personal digital assistance. However, due to its inherent data intensity of video sequences, storing and transmitting raw video data become impractical. With the limited storage and bandwidth capacity, this data must be compressed to a transportable size and finally decompressed to reach the destination. For this purpose, they studied and implemented two well known video compression and decompression algorithms i.e. Run Length Encoding (RLE) and Arithmetic Coding (AC). The main advantage of these compression techniques is that it almost compresses and decompresses all types of video files. In their work, they computed the compression rate on different video files by applying both algorithms. Experimentally, they find that AC algorithm compressed better in almost all types of video files except DAT type video file, and they also find that the compression rate of DAT file is highest and RLE performed better in it.

In view of Pereira, Fernando [PF 2011], Video compression is a key technology in the current multimedia services and applications landscape, which has been evolving to provide increasingly powerful user experiences. This evolution has continuously enlarged the set of networks and terminals able to provide video enabled experiences, as well as continuously improved the quality of the experiences provided for important services and applications, ranging from digital TV, and mobile and Internet video streaming to video games and Bluray discs. This evolution and its market impact is strongly determined by the set of video compression standards developed along the years since these standards allowed to provide easier interoperability and reduce the deployment costs. In their paper reviews the current status quo in video compression, as well as the main trends, with special emphasis on video compression standards, considering their particular influence on the deployment success of this technology.

According to Gaoture, Megha S., and Trupti H. Nagrare. [GT 2014], Nowadays, Video Compression is gaining importance because of its application in different fields like internet. So, there is different algorithm used to achieve the compression of Video. Video Compression algorithms combine spatial image compression and temporal motion compression. The sequence of frames contains Spatial and Temporal redundancies that Video Compression algorithms attempt to eliminate or code in smaller size. In their paper, they present a design of Binary motion vector technique with Pruning Discrete Wavelet Transform for the compression of AVI video. Here, Binary motion vector technique used for searching the Best matching block. This technique requires less candidates block than other motion vector technique. Pruning based DWT uses thresholding which will enhance the compression ratio with desirable Peak Signal to Noise ratio. This algorithm will be going to simulate on XilinxISE13.1 and implement on SPARTEN3 FPGA.

In view of Wang, Jian, and Yinghui Song, The video compression system is a key component of video data transfer system in the unmanned aerial vehicles (UAV). To satisfy the requirement of the wireless video transfer, their paper puts forward a cost effective, power efficient and low profile video compression schemes based on the H.264 technology. The video compression system consisted of a video decoder module based on TVP5150, an H.264 CODEC based on the i.MX27, an video encoder module based on CH7024, memory, external control module and power supply unit. Later the video compression software was introduced also. The video compression system cannot perform well in the flying experiment and the flying experiment proved the correction and robustness of the system.

According to Glaister, Jeffrey, Calvin Chan et.al. [GC+2011], their paper details a novel video compression pipeline using selective keyframe identification to encode video and patch-based super-resolution to decode for playback. Selective keyframe identification uses shot boundary detection and frame differencing methods to identify representative frames which are subsequently kept in high resolution within the compressed container. All other non-keyframes are downscaled for compression purposes. Patch-based super-resolution finds similar patches between an up scaled non-keyframe and the associated, high-resolution keyframe to regain lost detail via a super-resolution process. The algorithm was integrated into the H.264 video compression pipeline tested on webcam, cartoon and live-action video for both streaming and storage purposes. Experimental results show that the proposed hybrid video compression pipeline successfully achieved higher compression ratios than standard H.264, while achieving superior video quality than low resolution H.264 at similar compression ratios.

According to Wang, Shiqi, et al. [WS+2012], Compound video compression is crucial for remote control and data assessment. In their paper, they propose a content- aware layered video coding scheme as an attempt to efficiently compress the compound video. In that scheme, the compound video is analyzed and processed progressively at three pyramid levels: block, object and layer. Firstly, the compound video is analyzed by a block type classification technique to access each block’s spatial and temporal properties. Secondly, the natural video object is detected adaptively in each frame based on the block type. Finally, the compound video content is distributed into different layers and specifically designed video coding algorithms are employed to compress each layer. Experiments demonstrate that their proposed scheme can preserve the advantages of the employed compression algorithms for each layer and outperform each of them in the compound video compression.

The Authors Jeyakumar S., and S. Sundaravadivelu [JS 2008] say that, Video image compression has been an area where the computational demand is far above the capacity of conventional sequential processing. In their paper, they present a parallel motion estimation model for video sequence compression using distributed computing on a local network. The approach proposed is the decomposition of functions and data on a cluster of workstations using MPI mechanism. Parallel compression is achieved by having a multiple networked personal computer systems that perform compression on different chunks of input frames simultaneously. The method used for video compression is conventional block based motion vector estimation and a refined motion vector approximation that uses less side information for decoding. The implementation result shows that the proposed parallel method has better speedup than sequential algorithm and is very much suitable for real time applications like online video surveillance, video conferencing and telemedicine.

According to Ling, Nam [LN, 2010], in the near future, significantly increase in resolution and perceptual quality will be expected for home and mobile video applications. With future devices and content moving toward high-definition (for mobile applications) and ultra-high definition (for home applications), current AVC/H.264 technology will soon be unable to efficiently meet the compression demand for transmission. With the recent work from ISO/IEC MPEG high-performance video coding (HVC) and ITU-T VCEG groups, and the establishment of the Joint Collaborative Team (JCT), a new generation of video compression standardization process aiming at major improvements over the current AVC/H.264 standard has begun. In their paper, they briefly look at the technology and applications of video compression from the past to the present, and into possible directions and challenges for the next five years. They discuss key compression techniques and trends for future video coding efficiency, perceptual quality, and computational complexity.