www.ierjournal.org International Engineering Research Journal (IERJ) Volume 1 Issue 9 Page 728-731, 2015, ISSN 2395-1621

ISSN 2395-1621

# Video Compression Using FPGA

#1Sonali M. Wararkar, #2Prof. V.M.Joshi

#1sonali.wararkar13@gmail.com
#2joshivaishali7@gmail.com

#12Department of E&TC, PVPIT, Pune, India.

## ABSTRACT

**Video compression is a very useful tool to help eliminate the amount of resource video files consume. Compression creates a new file that stores data in a format that require less space. Video compression is reducing redundancy in video data. Compression technique used in video conferencing, video telephony, video on mobile phone, video on internet, HDTV broadcast. In this proposed work we will focus on compression and decompression of video. One of the common method of video compression is Discrete cosine Transform i.e. (DCT). In this paper we are taking one video of any byte then we generate a code for video compression and then we can see the result i.e. we can see reduced byte size of that video. This proposed work is design by using VHDL and system generator.**

*Keywords—* **Video compression, DCT, Simulink, Xilinx System Generator (XSG), Hardware Co-Simulation**.

Received:1st October 2015
Received in revised form :
2nd October 2015
Accepted:8th October, 2015
**Published online :**
**14th October 2015**

## I. INTRODUCTION

Videos in its original representation carries huge amount of data. Thus, it requires large amount of memory for storage. Video compression is an important area in image processing which efficiently removes the visually insignificant data. Compressed videos are sent over limited bandwidth channel with some additional processing for robust (error free) transmission. The 8x8 block-wise two-dimensional discrete cosine transform (2-D DCT) is used as orthogonal transform in video compression. Video compressed by this standard are used globally. This algorithm provides the user to choose between amount of compression and quality as per the requirement of the video in different applications. The variable amount of compression makes this algorithm very much suitable for the transmission purpose as user can adjust the bit rate of the transmission according to channel capacity.

## II. REDUNDANCY CODING

To compress data, it is important to recognize redundancies in data, in the form of coding redundancy, inter-pixel redundancy, and psycho-visual redundancy. Data redundancies occur when unnecessary data is used to represent source information. Compression is achieved when one or more of these types of redundancies are reduced. Intuitively, removing unnecessary data will decrease the size of the data, without losing any important information. However, this is not the case for psycho-visual redundancy.
The most obvious way to reduce compression is to reduce the coding redundancy. This is referring to the entropy of an image in the sense that more data is used than necessary to convey the information. Lossless redundancy removal compression techniques are classified as entropy coding. Other compression can be obtained through inter-pixel redundancy removal. In order to produce error free compression, it is recommended that only coding redundancy is reduced or eliminated. This means that the

source image will be exactly the same as the decompressed image.[1]

## III. THE HUMAN VISUAL SYSTEM

The Human Visual System (HVS) describes the way that the human eye processes an image, and relays it to the brain. By taking advantage of some properties of HVS, a lot of compression can be achieved. In general, the human eye is more sensitive to low frequency components, and the overall brightness, or luminance of the image.

Images contain both low frequency and high frequency components. Low frequencies correspond to slowly varying color, whereas high frequencies represent fine detail within the image. Intuitively, low frequencies are more important to create a good representation of an image. Higher frequencies can largely be ignored to a certain degree. The human eye is more sensitive to the luminance (brightness), than the chrominance(color difference) of an image. Thus during compression, chrominance values are less important and quantization can be used to reduce the amount of psycho-visual redundancy. Luminance data can be quantized, but more coarsely to ensure that important data is not lost. Several compression algorithms use transforms to change the image from pixel values representing color to frequencies dealing with light and dark of an image, not frequencies of light. Many forms of the JPEG compressions algorithm make use of the discrete cosine transform. Other transforms such as wavelets are employed by other compression algorithms. These models take advantage of subjective redundancy by exploiting the human visual system sensitivity to image characteristics.[1]

## IV. TYPES OF COMPRESSION

### 1. Lossless Compression

Lossless compression techniques work by removing redundant information as well as removing or reducing information that can be recreated during decompression. Lossless compression is ideal, as source data will be recreated without error. However, this leads to small compression ratios and will most likely not meet the needs of many applications. Compression ratios are highly dependent on input data, thus lossless compression will not meet the requirements of applications requiring a constant data rate or data size. [2][1]

Entropy encoders give a codeword to each piece of data. The codeword is of variable length to enhance compression. The varying length is typically determined by the frequency that a certain piece of data appears. Some algorithms generate code words after analyzing the data set, and others use standard code words already generated based off of average statistics. Shorter code words are assigned to those values appearing more frequently, where longer code words are assigned to those values that occur less frequently.

Run length coding, another form of entropy coding, was created to exploit the nature of inter-pixel redundancy. As each pixel is highly correlated to its neighbors, it can be expected that certain values will be repeated in adjacent pixels. By encoding this data repetition as a run length, significant compression can be achieved. When employed in lossy compression systems, run length coding can achieve significant compression compared to lossless compression

The best known lossless image compression algorithm is the CompuServe Graphics Interchange Format (GIF). Unfortunately the GIF format cannot handle compression of images with resolutions greater than 8-bits per pixel. Another lossless format called the Portable Network Graphics (PNG) can compress images with 24-bit or 48-bit color resolutions.[1]

### 2. Lossy Compression

The main benefit of lossy compression is that the data rate can be reduced. This is necessary as certain applications require high compression ratios along with accelerated data rates. Of course significant compression can be achieved simply by removing a lot of information and hence quality from the source image, but this is often inappropriate. Lossy compression algorithms attempt to maximize the benefits of compression ratio and bit rate, while minimizing loss of quality. Finding optimal ways to reach this goal is a severely complicated process as many factors must be taken into account. Lossy can give higher compression ratios than GIF while leaving the human observer with little to complain of loss in quality. Minimizing the variance between source and compressed image formats, otherwise known as Mean Square Error, is the ultimate goal for lossy compression algorithms.[2]

Compression induced loss in images can cause both missing image features as well as artifacts added to the picture. Artifacts are caused by noise produced by sources such as quantization, and may show up as blocking within the image. Blocking artifacts within an image can become apparent with higher compression ratios. The edges are representative of blocks used during compression, such as 8x8 blocks of pixels, used in JPEG. Strangely, artifacts such as noise or ringing may actually improve the subjective quality of the image. Noise can cause contouring which shows up in gradual shading regions of the image, while ringing is apparent at sharp edges. JPEG unfortunately does not do well with computer generated graphics. The proposed system uses this lossy compression technique.

## V. METHODOLOGY USED IN PROPOSEDNSYSTEM

Fig. 4.1 shows block diagram of the 2D DCT system

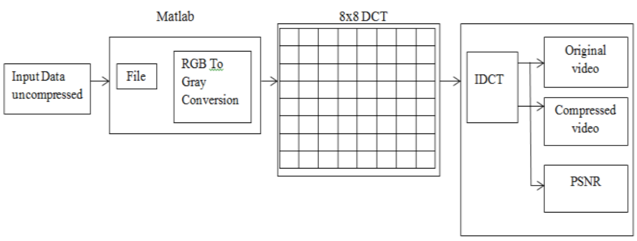



Fig. 4.1 Block diagram of 2D DCT

**Step 1. AVI File:** AVI means Audio Video Interleave. AVI files can contain both audio and video data in a file container that allows synchronous audio-with-video playback. AVI files can be created with no Compression, resulting in extremely large file sizes, but with no loss of quality from the input video to the saved file. This also requires no codec to be installed, either for saving or playback. This is generally not recommended.

**Step 2. RGB to Gray Conversion:** Transmitting images or video in RGB color space is not practical as their bandwidth requirement is very high. To overcome this problem and minimize the bandwidth requirement images in RGB color space are converted into other color space such as gray and then transmitted.[3]

**Step 3. Discrete Cosine Transform:** DCT is that popular in image processing is the ability of energy compaction.[4]

The formal definition of DCT is given below:

$$y(k) = w(k) \sum_{n-1}^{N} x(n) \cos\left(\frac{\prod(2n-1)(k-1)}{2N}\right) \quad k= 1,2\ldots..N$$

Where

$$w(k) = \begin{cases} \frac{1}{\sqrt{N}} & k = 1 \\ \sqrt{\frac{2}{N}} & = 2 \le k \le N \end{cases}$$

**Step 4. ZIGZAG PATTERN:** DCT coefficients are rearranged in increasing frequency order (zig-zag order).The first DCT coefficient is having zero frequency. It is called DC coefficient and the rest of the 63 coefficients are called AC coefficient. DC coefficient from the previous block are subtracted with the current block (differential coding). The DC coefficients represent the average image information of the block. The DC differential coding is performed to reduce the code size as nearest block possess the almost same average energy.[5]

**Step 5. INVERSE DISCRETE COSINE TRANSFORM:**

The inverse discrete cosine transform reconstructs a sequence from its discrete cosine transform (DCT) coefficients. The IDCT function is the inverse of the DCT function. x = IDCT(y) returns the inverse discrete cosine transform of y

$$x(n) = \sum_{k=1}^{N} w(k)\, y(k) \cos\left(\frac{\pi(2n-1)(k-1)}{2N}\right) \quad n = 1,2,\ldots\ldots N \qquad \ldots\ldots\ldots\ldots\ldots\ldots 4.4$$

Where

$$w(k) = \begin{cases} \frac{1}{\sqrt{N}} & k = 1 \\ \sqrt{\frac{2}{N}} & 2 \le k \le N \end{cases}$$

## VI. HARDWARE AND SOFTWARE IMPLEMENTATION

The software and hardware requirements of the system are mentioned below.

- Windows XP/windows 7
- MATLAB 13a
- Xilinx ISE 14.6
- Spartan-6 FPGA board
- Xilinx USB downloads cable.

## VII. RESULT AND DISCUSSION

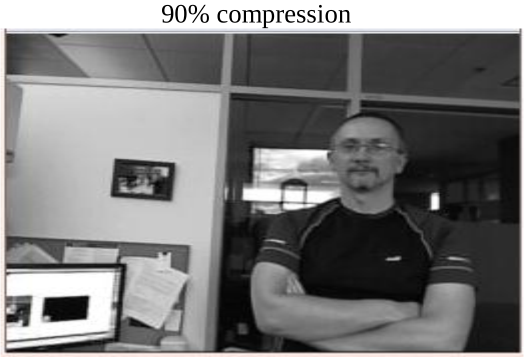
Table 1 Device utilization summary

| Device Utilization Summary (estimated values) | | | [-] |
|---|---|---|---|
| **Logic Utilization** | **Used** | **Available** | **Utilization** |
| Number of Slice Registers | 520 | 18224 | 2% |
| Number of Slice LUTs | 449 | 9112 | 4% |
| Number of fully used LUT-FF pairs | 119 | 850 | 14% |
| Number of bonded IOBs | 44 | 232 | 18% |
| Number of Block RAM/FIFO | 2 | 32 | 6% |
| Number of BUFG/BUFGCTRLs | 1 | 16 | 6% |
| Number of DSP48A1s | 10 | 32 | 31% |

Minimum period: 16.400ns (Maximum Frequency: 60.976MHz)
Minimum input arrival time before clock: 3.305ns
Maximum output required time after clock: 7.233ns
Maximum combinational path delay: No path found

Original video

90% compression



50% compression

10% compression

By the study of compression it shows that Discrete Wavelet Transform (DWT) performs compression of video in large amount and because of such huge amount of compression the quality of video/image reduced. But in Discrete Cosine transform (DCT) compression, quality of video is maintain even after performing the compression and this is the advantage of DCT compression

## VIII. CONCLUSION AND FUTURE SCOPE

Video-compression is reducing redundancy in video data Compression technique. Also allows more efficient storage & transmission of data. Xilinx software is used for both as synthesis and simulation for giving output result respectively. Verilog HDL is a hardware description language that can be used to model a digital system. By using DCT computation perform a large number of multiplication & addition but with well established and regular data access pattern DCT used which is simple to computation as compared to DWT, After reordering the reaming coefficients it is cheaper in cost than other compression algorithm also Require less storage & easy to computation. So by using require less blocks, minimum costing & simple algorithm video compression are done. Discrete Wavelet Transform (DWT) performs compression of video in large amount and because of such huge amount of compression the quality of video/image reduced. The processing of 1D DCT is very slow. Dual tree DCT consume more time. By using 2D DCT the drawback of 1D DCT, DWT, dual tree DCT has been eliminated. 2D DCT system had improved the speed of the compression, efficiency of the video also it consume less time. The future work will be that the compression can be done by using new format of compression i.e. H.264.

## REFERENCES

[1] Thesis on JPEG Image Compression Using an FPGA by James Rosenthal December 2006

[2] International conference on telecommunication and computing [ICTC-13] ECE, kits, Ramtek India on"VHDL design using concept of system on chip for video compression" by SHWETA ARUN SARODE M.TEC [Electronics Engineering] Student, and PROF.S.M.KHARAD Prof. and Head of Department Of E&C Engineering. Kavikulguru Institute of Technology Science, Ramtek, Maharashtra, india

[3] International Journal of Computer Applications (0975 – 8887) Volume 7– No.2, September 2010 on "A Theory Based on Conversion of RGB image to Gray image" by Tarun Kumar Assistant Professor Computer Science and Engineering Department Vidya College of Engineering, Meerut (U.P)and Karun Verma

[4] International Journal of Computer Applications (0975 – 8887) Volume 65–No.10, March 2013 on "An Area Efficient Design of Fully Pipelined, 2D DCT, Quantizer and Zigzag JPEG Encoder using VHDL" by Durg Patidar, Jaikaran Singh, Mukesh Tiwari, Department of Electronics and Communication Engineering Sri Satya Sai Institute of Science and Technology, Sehore M.P. India

[5] Thesis on Efficient VLSI Architectures for Image Compression Algorithms by Vijay Kumar Sharma January 2012