# EE 596 – IMAGE AND VIDEO CODING MINI PROJECT

MADUSHAN H.M.P.S.

E/17/199

SEMESTER 07

24/07/2023

## CONTENTS

1. Introduction
    1.1. Background
    1.2. Objectives and Scope

2. Stage 1: Basic Image Compression
    2.1. Image Coding System Implementation
    2.2. Quality estimation
    2.3. Compression Ratio Adjustment
    2.4. Adaptive Bitrate Algorithm

3. Stage 2: Basic Video Compression
    3.1. Video Compression System

4. Stage 3: Improved Hybrid Video Codec
    4.1. Optimization of Quantization Process
    4.2. Optimizing the Coding Process

5. Conclusion
6. References

# 1. INTRODUCTION

## 1.1 Background

In recent years, telecommunication systems have seen rapid growth in capacity. However, transmitting multimedia streams, especially raw video data, remains a challenge as data rates often surpass network capabilities. Video and image compression algorithms aim to reduce the data to match network capacity. Most compression schemes used for multimedia are "lossy," allowing some information loss to achieve high compression ratios. This trade-off between quality and compression ratio enables adaptation to the transmission channel's capacity. Hybrid video coding is a widely used technique in modern video coding standards, such as H.26x, MPEG2/4, AVS, and HEVC.

This project aims to explore the fundamental operations of image coding and block-based video coding systems, as well as the Rate-Distortion optimization techniques employed in video coding. Our goal is to develop a simplified hybrid video codec that incorporates essential coding tools like discrete cosine transformation, quantization, prediction, and entropy coding. Through a comprehensive analysis of each tool's influence on the codec's performance, we seek to gain insights into their roles in video compression and quality.

## 1.2 Objectives and Scope

- To investigate the basic functions of an image coding system.
- To investigate the basic functions of a block-based video coding system.
- To investigate the Rate -Distortion optimization techniques used in video coding.

Stage 1: Basic Image Compression: In this project, we will build a simple image coding system using discrete cosine transformation, quantization, and entropy encoding with varying levels for different output qualities. We will adjust the compression ratio to meet specific bit rates and create an algorithm for adaptive bitrate control.

Stage 2: Basic Video Compression: In this stage, we will expand the system to include video compression capabilities. We will integrate macro block-based coding, basic motion estimation, and intra prediction techniques to efficiently process video frames.

Stage 3: Improved Hybrid Video Codec: Optimize the quantization process of the codec to facilitate transmission in fixed bandwidth environment. We use an optimization analysis to find the best QP to meet the given bit-rate. Optimize the coding process to give the best bit-rate for a given quality level. You may use Intra and Inter prediction and feedback loop from the expected output quality to the quantization and the prediction stage to analyze the performance.

The goal of this study is to better understand hybrid video coding methods, how they are used, and how they affect video compression efficiency.

# 2. Stage 1: Basic Implementation: Image Compression

## 2.1 Image Coding System Implementation

The following steps are implemented in the GRAY scaled Lena Image to demonstrate the implementation of image compression system.

Color Space conversion

- Human eyes are more receptive to luminance (brightness and darkness) and less receptive to chrominance (color).
- Load RGB image and convert it to gray scaled image using `rgb2gray` function.

Discrete Cosine Transform

- Human vision is insensitive to high spatial frequencies.
- Divide image into 8x8 macro blocks which conations 64 pixels values, with 0-255 range using function `macroblocks`.
- Apply DCT transform to each 8x8 macro blocks using `DCT` function.

Quantization

- Divide each value in each macro block by Quantization matrix which multiplied with different quantize level values using `Quantize` function.
- Rounded off each result to nearest integer.

AC and DC values separation

- Extract the DC parts which is the first element in the Quantized coefficients matrices at upper left conner using the function `DC_Extract`
- AC coefficients were extracted using Zig-Zag scanning using the function `AC_Extract`
- Perform Differential coding on DC components using `differential` function.

Huffman Encoding

- Calculate AC/DC symbol probability and generate AC/DC Huffman codebooks using `huffmandict` function.
- Apply Huffman coding for AC and DC components separately using `huffmanenco` function.
- Save encoded AC and DC data as a bit stream.

Decoding

- Apply Huffman decoding for both AC & DC components separately using `huffmandeco`
- Do inverse Differential coding for DC elements using `inv_differential`
- Do inverse Run length coding for AC elements using `inv_runlength`
- Combine all Decoded AC & DC elements to generate macro blocks.
- Apply inverse Quantization process which involves multiply by the Quantization matrix and with the corresponding Quantization level using `inv_quantize`

- Apply inverse DCT transform to each macro block using `inv_DCT`
- Combine all macro blocks to generate one cell array which gives the decoded image using `inv_macroblocks`

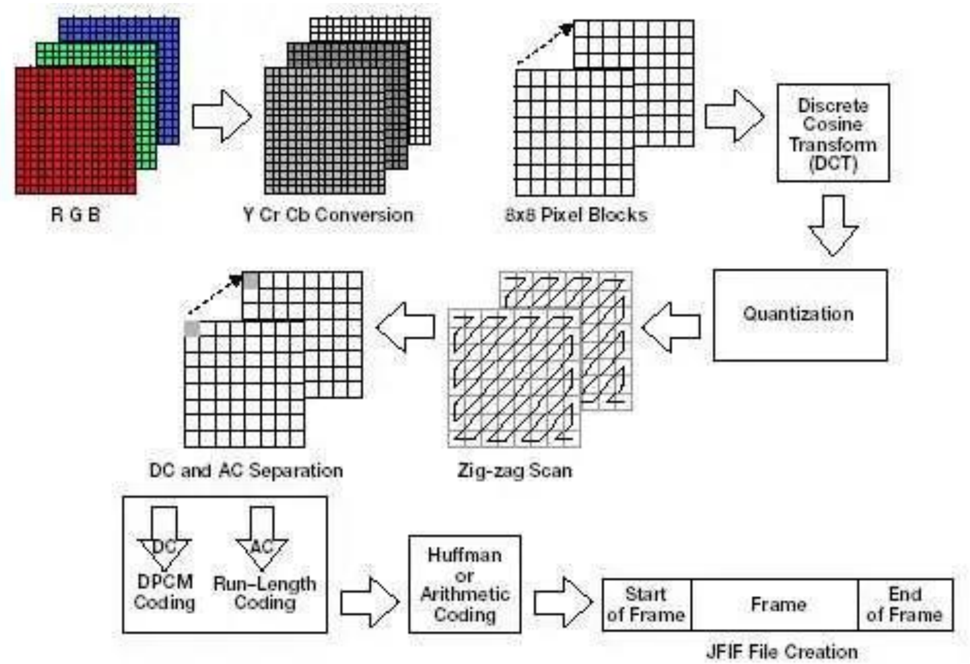


Figure 01: JPEG Image compression system

(a)

(b)

Figure 02: (a) Original Lena image (b) Grayscale Lena image

Different quality outputs are achieved by changing Q_Level in equation (1). When Q_Level is equal to one, new Quantization matrix is the recommended JPEG normalization matrix which identify the importance of values in DCT transformed data with respect to image quality.

$$Qnew = QLevel \cdot Q \quad \text{------ (1)}$$

No. of quantization levels are reduced when QLevel is high as explained by the equation (2). Then, QLevels of 4, 8, 16, 32 are used for quantization process and compressed images are as shown in Figure 02.

$$\widehat{\theta_{k,j}} = Q[\theta_{k,j}] = round\left(\frac{\theta_{k,j}}{Q_{new}}\right) \text{ --------(2)}$$

```
quantizationMatrix = QLevel* [16 11 10 16 24 40 51 61;
                              12 12 14 19 26 58 60 55;
                              14 13 16 24 40 57 69 56;
                              14 17 22 29 51 87 80 62;
                              18 22 37 56 68 109 103 77;
                              24 35 55 64 81 104 113 92;
                              49 64 78 87 103 121 120 101;
                              72 92 95 98 112 100 103 99];
```

Figure 03: Quantization Matrix

(a)

(b)

(c)

(d)

Figure 04: Decoded Lena image with different Q-Levels (a) QLevel = 4, (b) QLevel = 8, (c) QLevel = 16 and QLevel =32

## 2.2. Quality estimation

Table 01: Variation of Compression ratio and PSNR values with Q-level

| Quantization Level | Compression ratio | PSNR value | Quality |
|---|---|---|---|
| 4 | 2.93 | 31.26 | High |
| 8 | 5.2 | 28.46 | High |
| 16 | 10.06 | 24.9 | Medium |
| 32 | 23.69 | 20.95 | Low |

High Q-level will increase the quantization step size and results a coarser quantization. Coarse quantization leads to a higher levels of compression ratios. Information loss is increased and resulting lower image quality with lower PSNR value

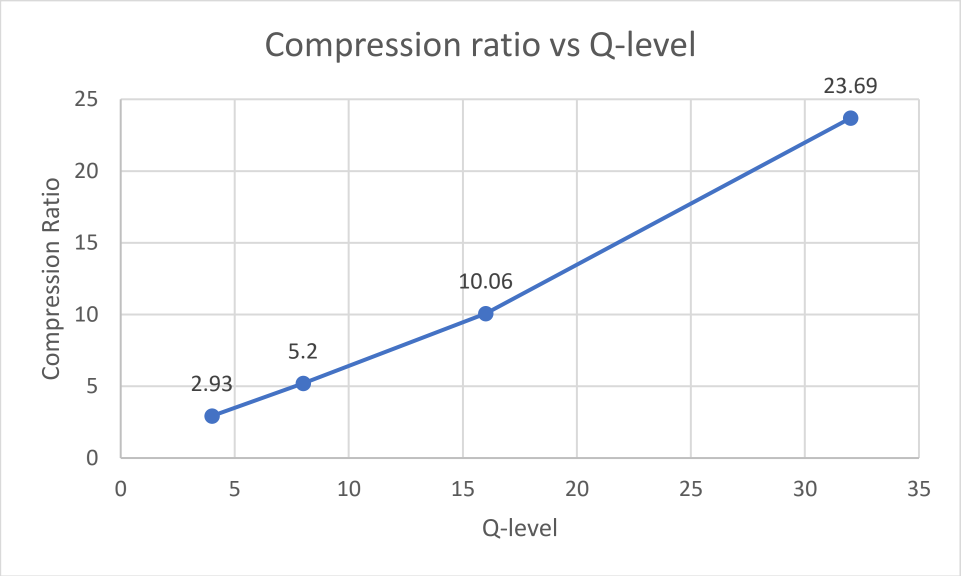


Figure 05: Variation of Compression ratio with Q-level

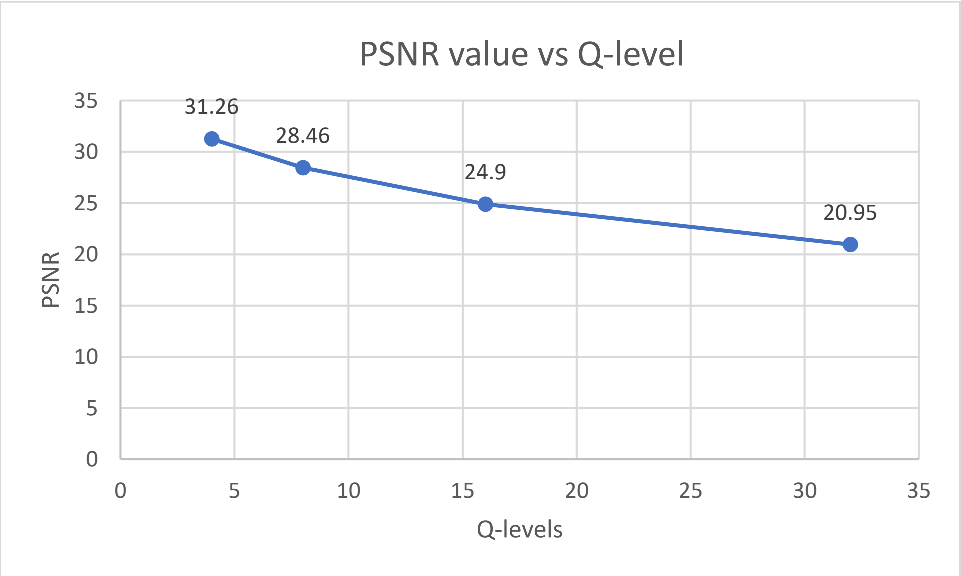


Figure 06: Variation of PSNR value with Q-level

### 2.3. Compression Ratio Adjustment

To achieve the desired bit rates, the image encoder's compression ratio was modified accordingly. To transmit one image per second, optimization was carried out to attain the optimal Peak Signal-to-Noise Ratio (PSNR) value, guaranteeing high-quality images within the specified bit rate.

| | |
|---|---|
| Bit rate of the channel | = 199 + 300 kbps = 499 kbps |
| Target Bit Rate per Image | = Total Bit Rate / Number of Images per Second |
| | = 499 kbps / 1 image/sec = 499 kb |
| Compression ratio (@499kb) | = 0.53 |
| Q –level | = 0.28388 |
| PSNR value | = 39.62 |

Table 02: Variation of Compression ratio and Bit rate values with Q-level

| Q- Level | Compression ratio | Bit rate(kb) |
|---|---|---|
| 8 | 5.2 | 50.445 |
| 4 | 2.93 | 89.584 |
| 2 | 1.78 | 147.08 |
| 1 | 1.13 | 232.726 |
| 0.5 | 0.73 | 358.816 |
| 0.25 | 0.49 | 536.778 |
| 0.285 | 0.53 | 497.978 |
| 0.28388 | 0.53 | 499.047 |

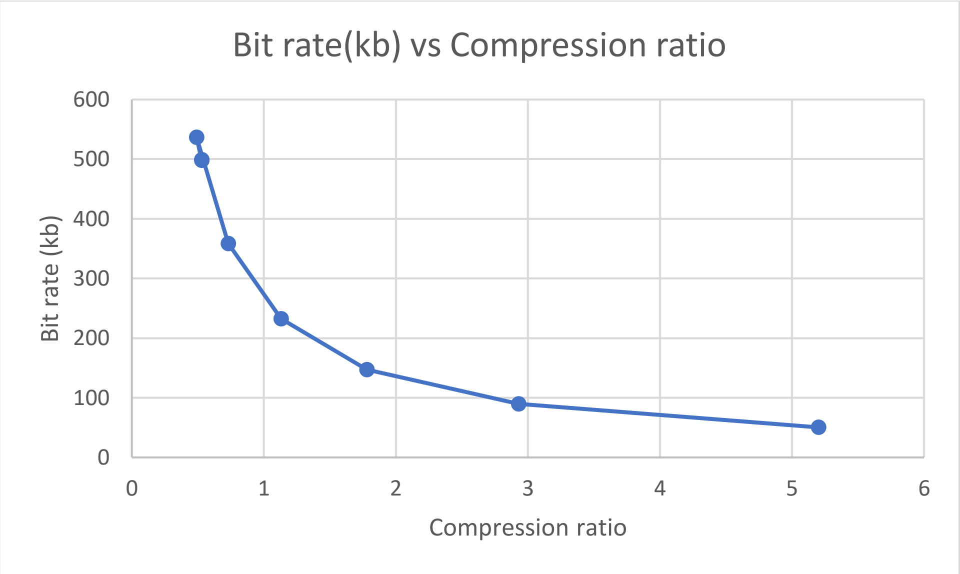


Figure 07: Variation of Bit rate with Compression ratio

Using this graph, you can optimize the quantization process by adjusting QLevel value to find the optimum QLevel value to meet the given bit-rate at given compression ratio value

# 3. Stage 2: Basic Video Compression

## 3.1. Video Compression System

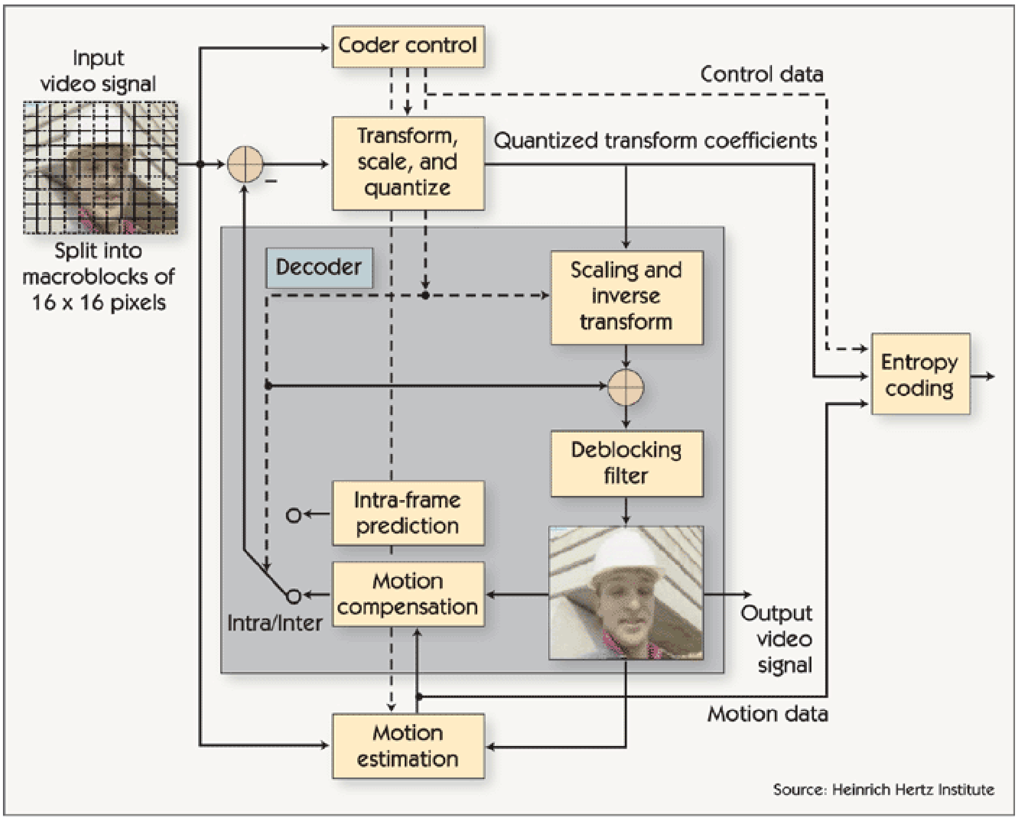


Fig. 2 H.264 macroblock coding structure.

Figure 08: Block Diagram of H264 Hybrid Video Codec

A video can be viewed as a series of frames, and each frame can be treated as an image compression system. By leveraging spatial redundancy, temporal redundancy, and source coding redundancy, we can enhance the compression ratio for lower bitrates, making them suitable for data storage and transmission purposes. To achieve this, motion estimation and intra prediction techniques are employed to eliminate redundancy. The compression system employs a GOP (Group of Pictures) structure, specifically IPPP. For motion estimation, the full search algorithm is utilized.

The video compression system goes through the following stages for 10 consecutive frames of a sample video, which is a GRAY scaled Big Buck Bunny footage.

1. Read video into frames and extract 10 consecutive frames.
2. Setup the Q-level, macro block size and search area parameters
3. Each frame is converted RGB to gray scale using `rgb2gray` function.
4. Divide each frame into 8x8 macro blocks using `macroblocks` function.
5. Group of Pictures (GOP) structure used is IPPP (the first frame in each group is an I-frame (Intra-coded frame), followed by three P-frames (Predictive-coded frames)
6. If I-frame, do encode by,
   - DCT Transform
   - Quantization

- AC/DC Extraction
- Differential coding on DC components
- Run length coding on AC components.
- Huffman coding for AC & DC components separately
- Store AC and DC encoded bit streams

7. If P-frame, do encode by
    - Motion estimation using full search algorithm is used to calculate Motion vector and Residual image between predicted P-frame and current P-frame using `motion_estimate` function.
    - Huffman encode the residual frame using same procedure as before (which is done to I-frame) using `encode_image` function.
    - Huffman encode Motion vectors using `encode_MV`.

8. Store the I-frame encoded AC/DC bit streams, P-frame Residual encoded AC/DC bit stream and motion vector encoded bit streams.
9. Decoding process, encoded I-frame bit streams are decoded,
    - Huffman decoding AC & DC components separately
    - Inverse Run length coding for AC components
    - Inverse Differential coding for DC components
    - Inverse extraction process of AC & DC components and rebuild the matrix.
    - Inverse Quantization
    - Inverse DCT transform.
10. Decoding process, encoded P-frames are decoded,
    - Huffman decoding is done to decode the encoded residual of P-frame and encoded motion vectors separately using `decode_MV`.
    - Inverse motion estimation is done to Recreate the P-frames using `Inv_Motion_Estimate`
11. Decoded output frame sequence is created by removing the macro blocks.

I P P P

Figure 09: Frame sequence for IPPP structure of first 4 original frames

I P P P

Figure 10: Compressed frame sequence for IPPP structure of first 4 Decoded frames with Q-level =8


Figure 11: Compressed frame sequence for IPPP structure of first 4 Decoded frames with Q-level = 16

Higher QLevels indicate more losses in the decoded image, leading to increased errors in the decoded image sequence. In Figure 11, these errors are more noticeable compared to Figure 09, primarily because of imperfections in motion prediction and intra prediction, resulting in larger quantization errors.

# 4. Stage 3: Improved Hybrid Video Codec

## 4.1. Optimization of Quantization Process

In this stage, the quantization process of the codec was optimized to facilitate transmission in a fixed bandwidth environment. An optimization analysis was performed to determine the best Quantization Parameter (QP) that met the given bit rate requirement while preserving acceptable video quality. The goal was to strike a balance between compression efficiency and perceptual video quality. The codec's bit-rate can be controlled by modifying the quantization parameter. Lower QP values indicate finer quantization, resulting in higher bit-rates and better video quality. Conversely, higher QP values lead to coarser quantization, resulting in lower bit-rates and potentially lower video quality.

Bit rate of the channel = 199 + 300 kbps = 499 kbps

Target Bit Rate per frame = Total Bit Rate / Number of frames per second

= 499 kbps / 30 frames/sec

= 16.663 kb per frame

Optimal Q –level value = 40

Figure 12: Compressed output frame sequence for fixed bit rate of 16.633kb per frame with Q-level = 40

## 4.2. Optimizing the Coding Process

The coding process is optimized to achieve the best bit-rate while maintaining a given quality level. This optimization involves utilizing Intra and Inter prediction techniques, along with a feedback loop from the expected output quality. By continuously analyzing the performance on the fly, the feedback loop informs adjustments to both quantization and prediction stages, ensuring an optimal balance between compression efficiency and video quality. This dynamic approach allows for real-time adaptation, leading to the most efficient coding process and the desired quality level.

Required PSNR quality level = 40

Q –level value = **2**

Maximum possible PSNR value = 44.2098

PSNR = 40.9037 PSNR = 41.4736 PSNR = 41.4376 PSNR = 41.1946

Figure 13: Compressed output frame sequence for fixed quality level of PSNR = 40 with Q-level = 2

## 5. CONCLUSION

The investigation and implementation of a simplified hybrid video codec, with a particular emphasis on image and video coding techniques and Rate-Distortion optimization, yielded successful outcomes. Throughout the project's different stages, valuable insights were gained concerning the hybrid video codec's functionalities and performance.

In Stage 1, a fundamental image coding system was effectively created, utilizing discrete cosine transformation and quantization. Stage 2 extended this system to support video compression by incorporating macro block-based coding, motion estimation, and intra prediction. This integration enabled efficient representation and compression of video content, effectively utilizing spatial and temporal redundancies.

During Stage 3, the codec's performance underwent optimization in multiple aspects. The quantization process was fine-tuned to ensure efficient transmission in fixed bandwidth environments while maintaining an acceptable level of video quality. Through the determination of the best Quantization Parameter (QP) values, an optimal trade-off between compression efficiency and perceptual video quality was achieved.

## 6. REFERENCES


- Hood, A. (2017) Image and Video Compression, Ppt Video Online Download.
  Available at: https://slideplayer.com/slide/5928829/
  (Accessed: 22/06/2023, 16:15)
- Image and Video Compression (2020).
  Available at: https://www.powershow.com/view4/4424ea-YTM5Y/Image_and_Video_Compression_powerpoint_ppt_presentation. /
  (Accessed: 23/06/2023, 08:45)
- Branch Education (2021) How are Images Compressed?  [46MB ↘↘ 4.07MB] JPEG In Depth.
  Available at: https://www.youtube.com/watch?v=Kv1Hiv3ox8I.
  (Accessed: 23/06/2023, 20:40)
- GeeksforGeeks (2023) "Huffman Coding   Greedy Algo 3," GeeksforGeeks [Preprint]. Available at: https://www.geeksforgeeks.org/huffman-coding-greedy-algo-3/. (Accessed: 24/06/2023, 09:50)
- Lee, J.-H., Shin, I. and Park, H. (2006) "Adaptive Intra-Frame Assignment and Bit-Rate Estimation for Variable GOP Length in H.264," IEEE Transactions on Circuits and Systems for Video Technology, 16(10), pp. 1271–1279.
  Available at: https://doi.org/10.1109/tcsvt.2006.881856.
  (Accessed: 25/06/2023, 23:35)