Video Compression Using Unsupervised Learning

Group-24: Aman Kumar (MT24012), Hardik Jain (MT24037), Suhail Saifi (MT24097)

Introduction

This project demonstrates video compression using techniques like block-based motion compensation, K-means clustering, and residual error encoding. The objective is to reduce video file size while maintaining visual quality, measured by Peak Signal-to-Noise Ratio (PSNR). This approach is applicable to scenarios where video storage or streaming is crucial.

The project follows a block-based approach, where each frame is divided into smaller blocks (16x16 pixels), making it easier to apply com-pression techniques. By using K-means clustering, each block is grouped into a cluster, reducing the amount of data required to represent the frame. Motion compensation techniques are then applied to estimate and encode the motion be-tween frames, ensuring efficient compression.

In the following sections, the video processing, compression, decompression, and performance evaluation of the methods are explained.

Keywords: *Video compression, motion compensation, K-means clustering, PSNR, residual encoding.*

Key Techniques and Methods

1. Video Capture and Initialization

The video is loaded from a specified path using OpenCV's cv2.VideoCapture(). Key parameters like block size (16x16) and the number of clusters (5) for the K-means algorithm are set.

A maximum of 60 seconds (1439 frames at 24 fps) is processed for demonstration.

2. Frame Processing and Block Division

Each frame is divided into 16x16 pixel blocks, allowing better handling and compression of in-dividual blocks. This reduces the complexity of encoding the entire frame at once.

3. K-means Clustering

Each block's average colour is calculated, and K-means clustering is applied to group blocks into 5 clusters. The result helps to simplify the encoding of each block, reducing the amount of data needed to represent the video.

4. Motion Compensation

Motion vectors are computed between consecutive frames using a diamond search algorithm. This technique compares blocks in the current frame with those in the reference frame to find the best match, encoding the motion between frames.

Fig 1: Motion compensation

5. Compression

For each frame, the motion vectors are calculated, and the compressed data includes:

- Cluster assignments: Identifies which cluster each block belongs to.
- Motion model: Describes the motion vectors.
- Block size and frame shape: Needed for re-construction.
- Residual frame: The difference between the original and predicted frame, com-pressed as well.

Fig 2: Residual Frame

6. Decompression

During decompression, the motion vectors predict the content of a frame based on the reference frame. The residual frame is then added back to reconstruct the final frame. The process repeats for each frame.

7. Refresh Mechanism

A refresh mechanism is used to periodically replace the reference frame after a certain number of frames (4 in this case). This helps to reduce the accumulated error and optimize the compression and decompression process.

Performance Evaluation

SNR Calculation: Peak Signal-to-Noise Ratio (PSNR) is calculated for each frame by comparing the original and decompressed frames. The average PSNR provides a measure of the decompressed video's quality. A higher PSNR indicates better quality.

Results

The average PSNR values are calculated across all frames, showing the effectiveness of the com-pression method in maintaining visual quality. For this project, the SNR was satisfactory for maintaining high-quality video playback at a reduced size.

$$MSE = \frac{1}{m\,n}\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}[I(i,j)-K(i,j)]^2.$$

The PSNR (in dB) is defined as

$$\begin{aligned} PSNR &= 10\cdot\log_{10}\left(\frac{MAX_I^2}{MSE}\right) \\ &= 20\cdot\log_{10}\left(\frac{MAX_I}{\sqrt{MSE}}\right) \\ &= 20\cdot\log_{10}(MAX_I) - 10\cdot\log_{10}(MSE). \end{aligned}$$

Conclusion

This project presents a video compression approach using motion compensation, block-based compression, K-means clustering, and residual encoding. The techniques used are foundational to modern video codecs such as H.264 and HEVC, making this project an insightful learning experience in video compression theory. Potential improvements include experimenting with different block sizes, clustering methods, and motion compensation techniques to enhance compression efficiency.

REFERENCES

[1] Kaur, A., and R. Kaur. "Feature extraction from video data for indexing and retrieval." International Research Journal of Engineering and Technology (IRJET) 2, no. 1 (2015): 108-115.

[2] Pfaff, Jonathan, Heiko Schwarz, Detlev Marpe, Benjamin Bross, Santiago De-Luxán-Hernández, Philipp Helle, Christian R. Helmrich et al. "Video compression using generalized binary partitioning, trellis coded quantization, perceptually optimized encoding, and advanced prediction and transform coding." IEEE Transactions on Circuits and Systems for Video Technology 30, no. 5 (2019): 1281-1295.

[3] Maksimov, Aleksey, and Mikhail Gashnikov. "Generalization of Machine Learning-Based Image Compression Methods for Video Compression." In 2023 IX International Conference on Information Technology and Nanotechnology (ITNT), pp. 1-5. IEEE, 2023.

[4] Tang, Hao, Lei Ding, Songsong Wu, Bin Ren, Nicu Sebe, and Paolo Rota. "Deep unsupervised key frame extraction for efficient video classification." ACM Transactions on Multimedia Computing, Communications and Applications 19, no. 3 (2023): 1-17.

[5] Lampert, Christoph H. "Machine learning for video compression: Macroblock mode decision." In 18th International Conference on Pattern Recognition (ICPR'06), vol. 1, pp. 936-940. IEEE, 2006.