MATHEMATICS OF INTELLIGENT SYSTEMS-5

21 MAT 301

TEAM-MEMEBRS

1.JAIDEV . K – CB.EN.U4AIE21117

2. SAI CHANDANA . J – CB.EN.U4AIE21118

3.PRANISH KUMAR. M – CB.EN.U4AIE21137

4. CHARISHMA CHOWDARY .T - CB.EN.U4AIE211169

CONTRIBUTIONS

1.JAIDEV. K – MOTION ANALYSIS & VIDEO COMPRESSION

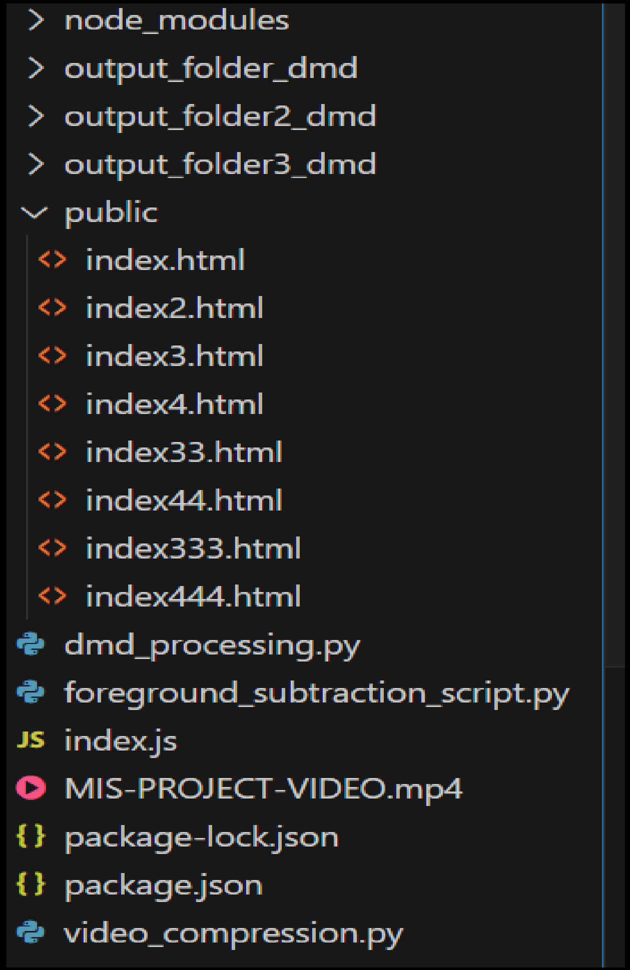
2.SAI CHANDANA J – MOTION ANALYSIS & FOREGROUND SUBTRACTION

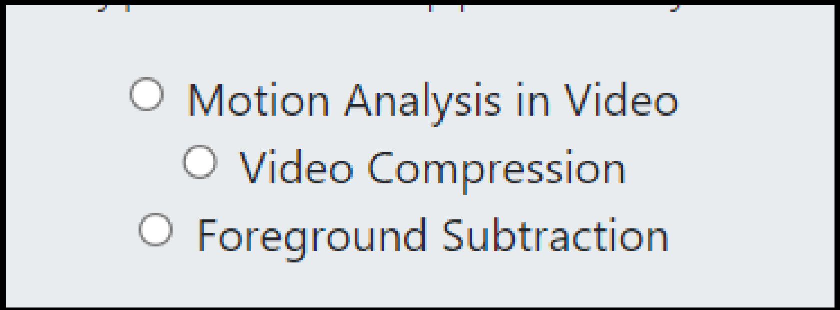
3.PRANISH KUMAR. M - MOTION ANALYSIS & VIDEO COMPRESSION

4.CHARISHMA CHOWDARY. T - MOTION ANALYSIS & FOREGROUND SUBTRACTION

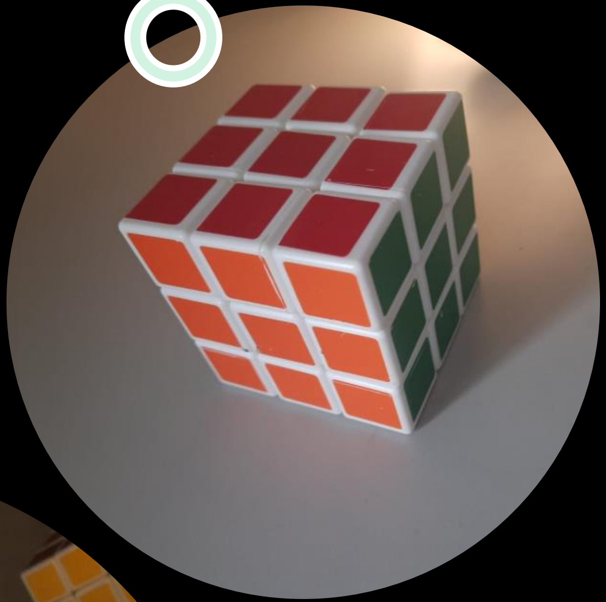
VIDEO ANALYSIS USING DMD

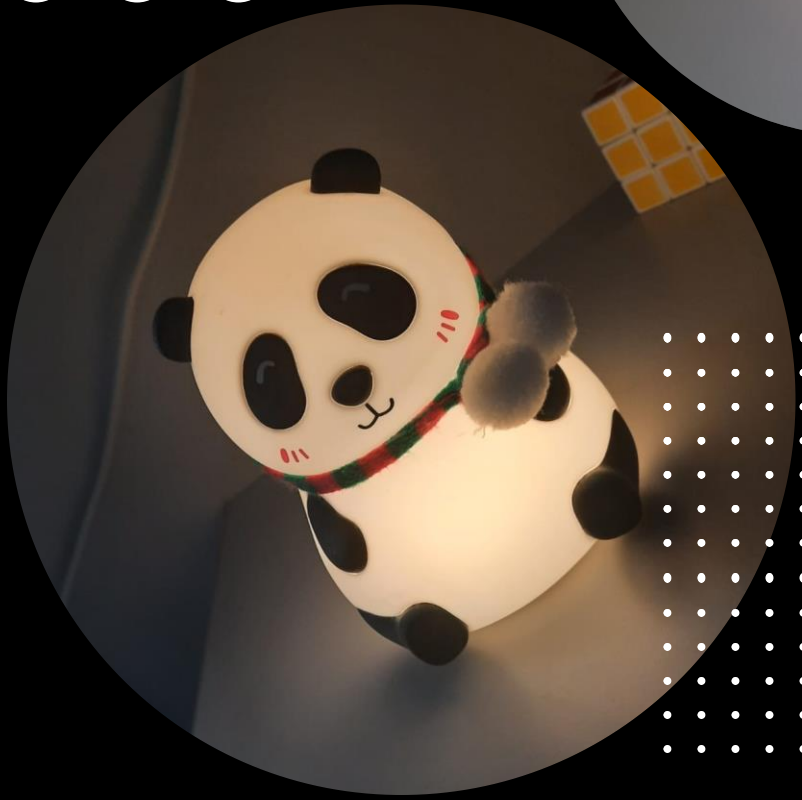
Files in our project



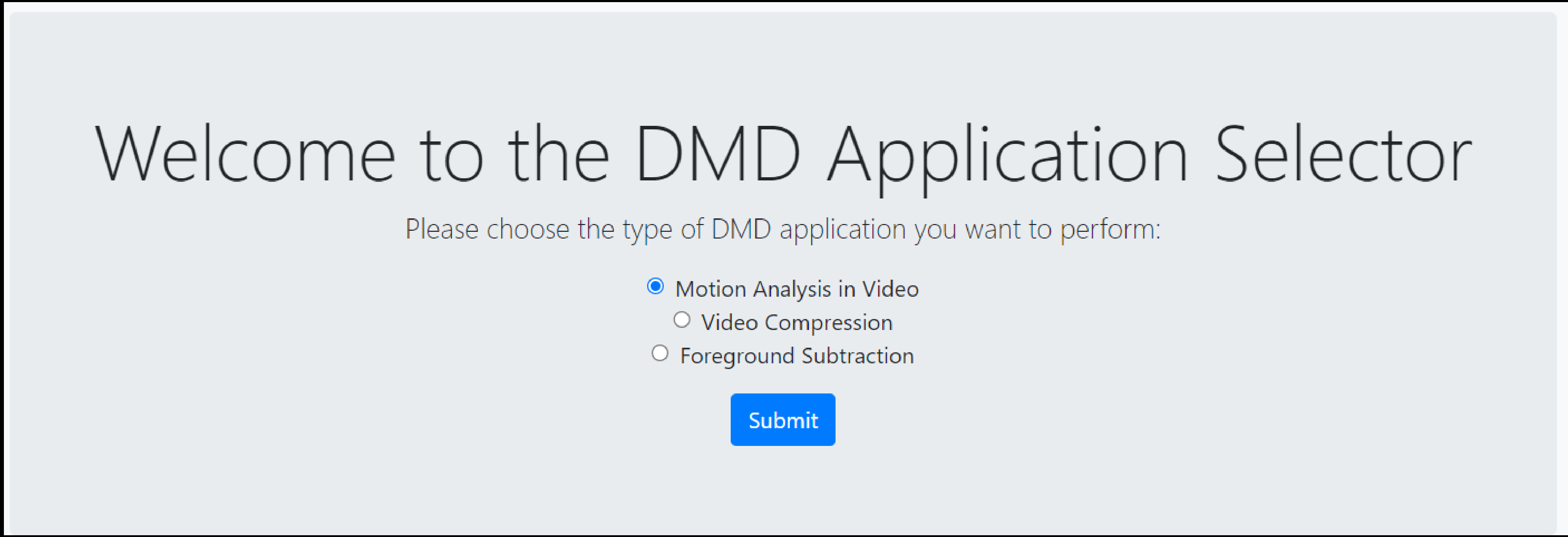


➢ MATERIALS USED IN OUR PROJECT



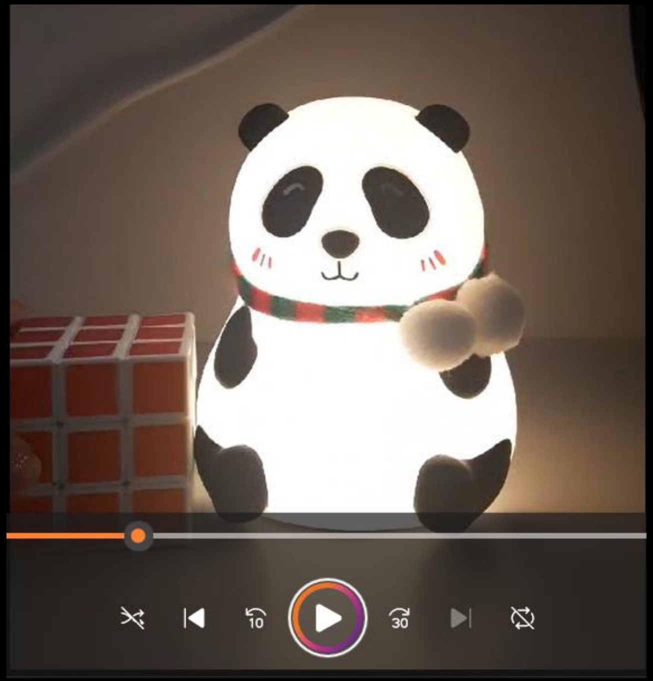


GRAPHICAL – USER-INTERFACE



MOTION ANALYSIS IN VIDEO

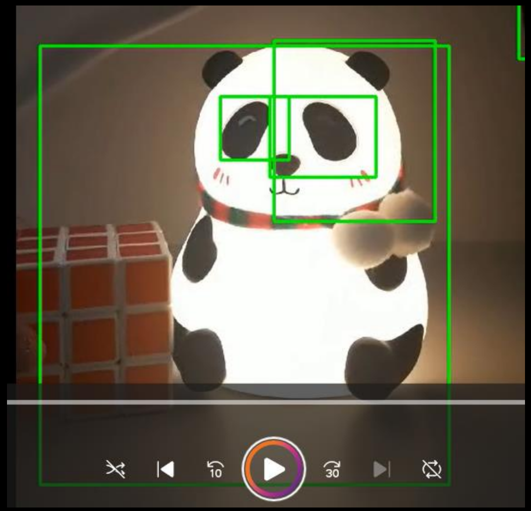
MOTION ANALYSIS IN VIDEO

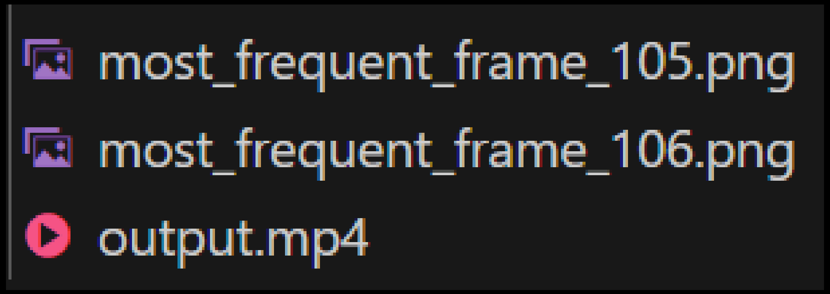


1.INPUT - MIS-PROJECT-VIDEO

OUTPUT

- motion analyzed video
- least frequently changed frame
- most frequently changed frame





WORKING

- **DMD Initialization:**

  DMD is applied to analyze the dynamic behavior of the video frames.

  An instance of the **DMD** class from the **pydmd** library is created.

- **Singular Value Decomposition (SVD) Rank:**

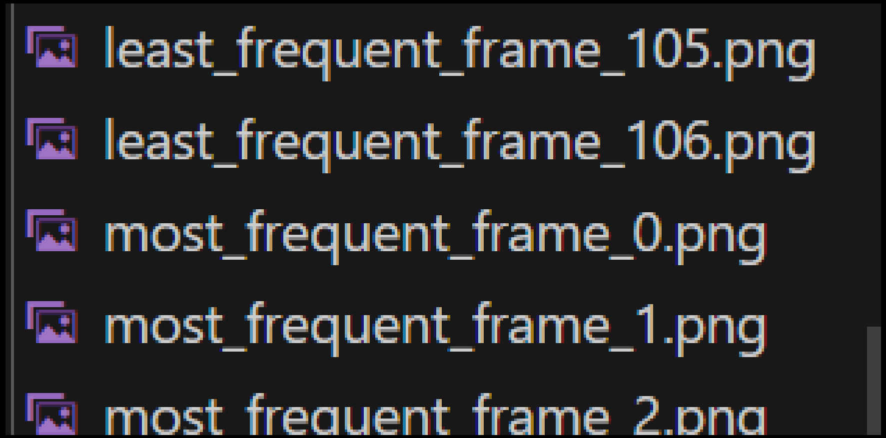
  The parameter **svd_rank** is set to 5 when initializing the DMD instance.

  This parameter controls the number of singular values and vectors retained during the decomposition, affecting the complexity of the decomposition.

➢frame Differences for DMD:

The absolute difference between consecutive frames is calculated using OpenCV.

This difference represents the changes in pixel values between frames.



- **DMD Fitting:**

  The DMD algorithm is applied to fit the absolute difference frames.

  DMD decomposes the input frames into spatial modes and corresponding temporal dynamics.

- **Modes and Dynamics:**

  The decomposition results in spatial modes and temporal dynamics.

  Spatial modes represent the spatial patterns of motion in the frames.

  Temporal dynamics represent how each mode evolves over time.

- **Background Reconstruction:**

  The product of spatial modes and temporal dynamics is used to reconstruct the background of the video.

  This background represents the low-rank approximation of the video frames.

- **Foreground Extraction:**

  The difference between the absolute difference frame and the reconstructed background is taken.

  This process aims to highlight the moving or changing elements in the video.

- **Thresholding:**

  A threshold is applied to the difference to create a binary mask.

  Pixels with values above the threshold are considered significant changes.

- **Contour Detection on Foreground:**

  Contours are identified in the binary mask, outlining regions of significant change.

  These contours correspond to moving objects or areas with notable differences between frames

- **Post-Processing:**

  Post-processing steps may include filtering contours based on size or other characteristics to refine the detection of moving objects.

- **Integration with Video Processing Loop:**

  The DMD process is repeated for each pair of consecutive frames in the video.

  The detected changes are then used for object tracking and frequency analysis.

TO VIEW OUR PROJECT

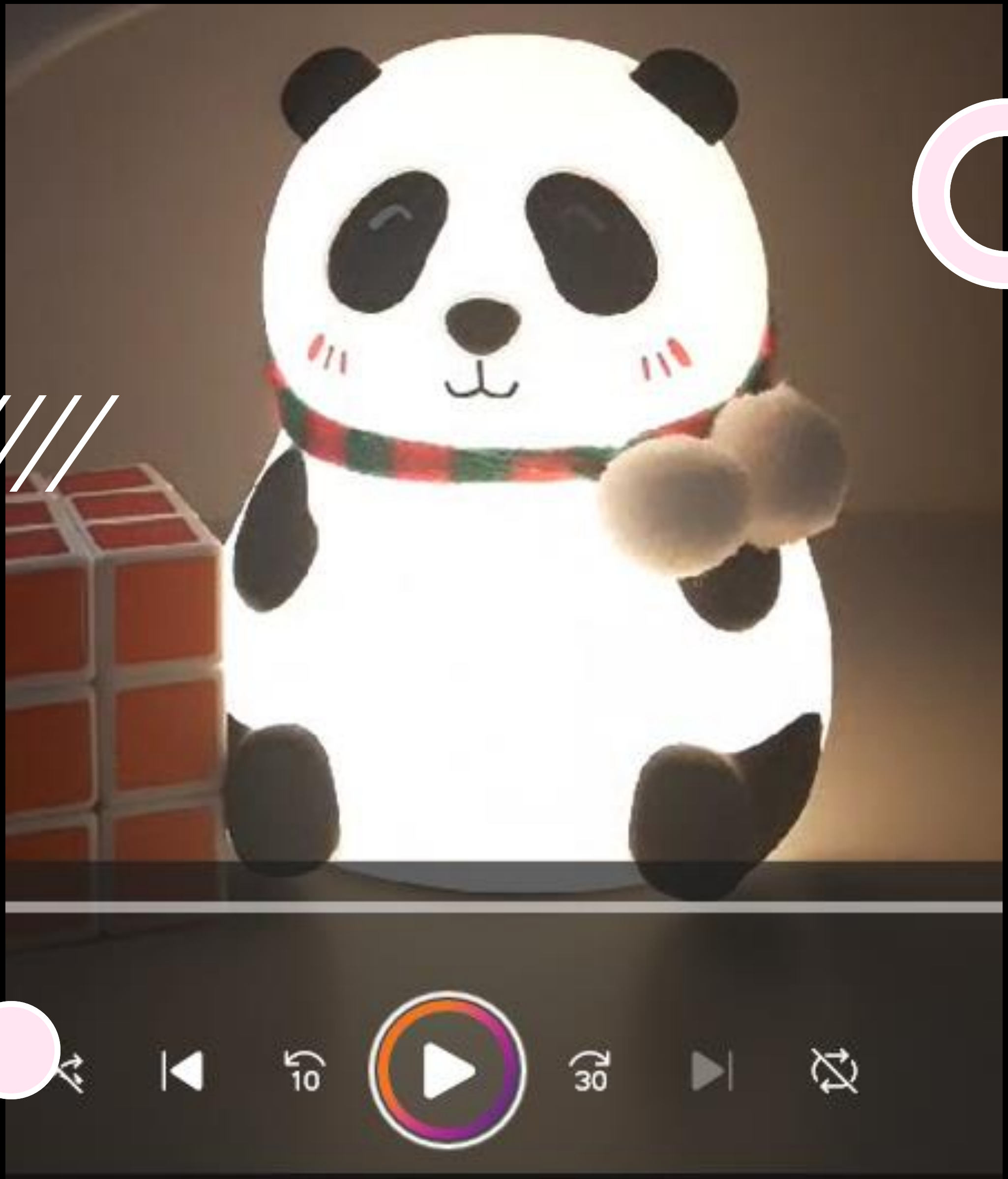
Code Repository -
https://github.com/sai-chandana-jampala/MIS-END-SEMESTER-PROJECT

Our website -
http://localhost:3000/

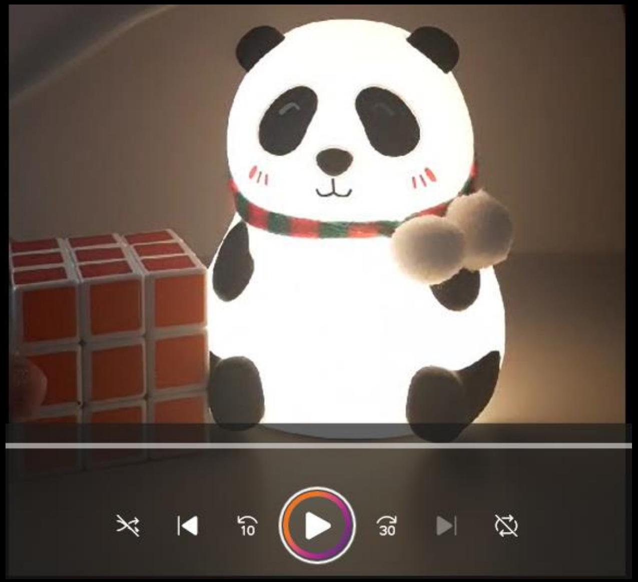
VIDEO COMPRESSION

VIDEO COMPRESSION

➢INPUT - MIS-PROJECT-VIDEO



➢ <u>OUTPUT</u> – COMPRESSED VIDEO



```
Python script output: Original video size: 0.77 MB
PSNR between original and reconstructed videos: 38.75 dB
Compressed video size: 0.41 MB
```

WORKING

- **1.Video Compression Overview:**

  The script **video_compression.py** focuses on compressing a video using the Dynamic Mode Decomposition (DMD) technique.

- **2. Reading and Storing Frames:**

  The function **video_to_frames** reads each frame from the input video and stores them in a list.

  This list of frames is later used for compression.

- **3. Writing Frames to Video:**

  The function **frames_to_video** takes a list of frames and writes them to an output video file.

  It uses OpenCV's **VideoWriter** to create the compressed video.

- **PSNR Calculation:**

  Peak Signal-to-Noise Ratio (PSNR) is calculated using the mean squared error between the original and reconstructed videos.

  PSNR provides a quantitative measure of the quality of the compressed video.

- **DMD Compression Process:**

  DMD is a key component for video compression in this script.

  DMD is applied to the flattened 3D array of frames (height, width, time).

  It decomposes the frames into spatial modes and temporal dynamics.

- **Original and Reconstructed Videos:**

  The script prints the size of the original video and converts the frames into a 3D array for DMD processing.

  DMD is used to reconstruct the video, and the PSNR between the original and reconstructed videos is calculated.

- **Reshaping and Normalization:**

  Reconstructed frames are reshaped and normalized to ensure pixel values are within the valid range (0-255).

  The reconstructed frames are clipped to this range and converted to the **uint8** data type.

- **Compressed Video Saving:**

The compressed video is saved with the appropriate file extension using the **frames_to_video** function.

- **Command-Line Arguments:**

The script accepts command-line arguments for the input video path, output video path, and an optional rank for the DMD process.

- **DMD Rank:** - The rank parameter in DMD controls the number of singular values and vectors retained during the decomposition. - It impacts the quality and complexity of the compression.

- **Printing Compression Results:** - The script prints the PSNR value between the original and reconstructed videos. - It also displays the sizes of the original and compressed videos.

TO VIEW OUR PROJECT

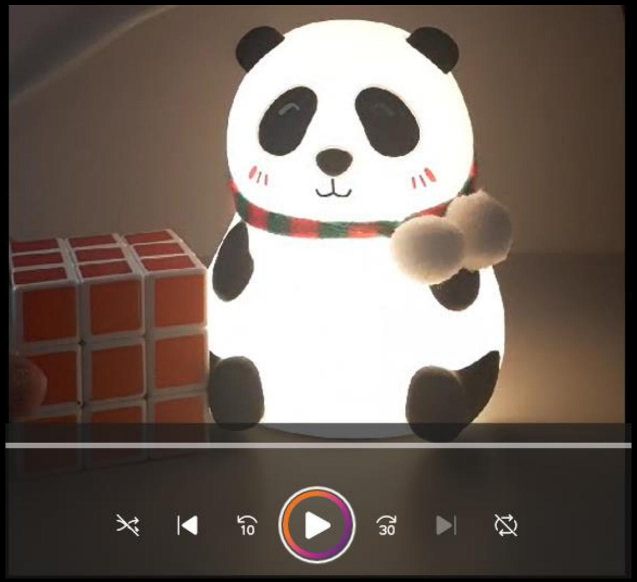
Code Repository -
https://github.com/sai-chandana-jampala/MIS-END-SEMESTER-PROJECT

Our website -
http://localhost:3000/

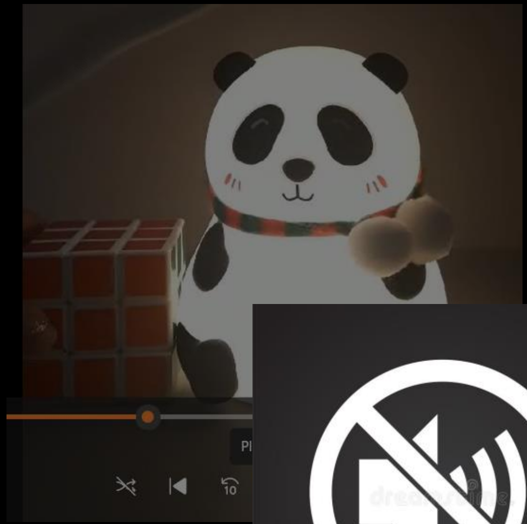
FORGROUND SUBSTRACTION

FORGROUND SUBSTRACTION

- Input – mis-project video



Output – foreground video (with out noise)



working

- **Video Background Removal Overview:**

  The script aims to remove the background from a video using Dynamic Mode Decomposition (DMD).

- **Video Reading and Conversion:**

  The script reads frames from the input video using OpenCV and converts them into a NumPy array.

  Frames are stored in a list, and the array is reshaped into 2D (space x time).

- **Dynamic Mode Decomposition (DMD):**

  DMD is applied to the reshaped video data with a specified Singular Value Decomposition (SVD) rank.

  The DMD algorithm decomposes the video into spatial modes and temporal dynamics.

- **Background Reconstruction:**

The script extracts the background by taking the real part of the first mode obtained from DMD.

This background represents the dominant spatial structure of the video.

- **Foreground Extraction:**

The foreground is obtained by subtracting the background from the original video data.

It represents the dynamic elements or moving objects in the video.

- **Reshaping Foreground:**

  The script reshapes the foreground to match the original shape of the video frames.

- **Output Folder Creation:**

  The script ensures that the specified output folder exists or creates it if not.

- **Saving Foreground Video:**

  The foreground video is saved as 'foreground.mp4' in the specified output folder.

```
Python script exited with code 0
Python script output: Foreground video saved at: C:\Users\HP\mis\output_folder3_dmd\output_video.mp4\foreground.mp4
```

TO VIEW OUR PROJECT

Code Repository -
https://github.com/sai-chandana-jampala/MIS-END-SEMESTER-PROJECT

Our website -
http://localhost:3000/

CONCLUSION

- The project collectively illustrate the versatile applications of Dynamic Mode Decomposition (DMD) in video processing. In the **dmd_object_tracking** script, DMD is employed for real-time object tracking, where it effectively highlights and tracks moving objects within a video. The **video_compression.py** script showcases how DMD can be harnessed for video compression, decomposing video frames into spatial modes and temporal dynamics to reconstruct a compressed video. Meanwhile, the **remove_background_dmd** script utilizes DMD for background removal, enabling the extraction of dynamic foreground elements from video frames. The common thread across these scripts is the effective use of DMD to analyze the dynamic behavior of video content.

INFERENCE

- The inference drawn from these Project is that Dynamic Mode Decomposition (DMD) serves as a powerful tool for video analysis and manipulation. Its ability to decompose video frames into spatial modes and temporal dynamics makes it adaptable to various tasks, ranging from object tracking to video compression and background removal. The incorporation of a graphical user interface (GUI) further enhances accessibility, providing users with an intuitive and user-friendly means to interact with these complex video processing functionalities. The versatility demonstrated by these scripts suggests that DMD could find valuable applications in fields such as surveillance, video compression, and computer vision, where efficient analysis and representation of dynamic content are essential. Overall, these scripts provide a glimpse into the potential of DMD as a valuable tool in the realm of video processing and analysis.

THANK YOU