Emotion Detection and Summarization in Videos

1.1)Emotion Detection Classification

Objective

The goal of this project is to create a machine learning model that can identify different human emotions from pictures of their faces. One of the seven fundamental emotions—anger, disgust, fear, happiness, neutrality, sadness, and surprise—must be identified in each image.

Methodology

Dataset: Made use of a dataset of emotions labelled on faces in images. Three sets of the dataset were created: training, validation, and test.

Preprocessing: To standardise the input for the model, images were first processed. Among them were:

- resizing pictures to maintain a standard size
- converting pixel values from 0 to 1,
- Increasing the size and variability of the training dataset by applying rotations, shifts, flips, and zooms.

Architecture Model: Convolutional Neural Networks (CNNs) were selected due to their superior performance in image classification applications. What made up the model architecture was:

- Convolutional layers are used to identify features in the pictures.
- Max pooling layers help preserve key features while reducing dimensionality.
- To turn feature maps into a 1D vector, flatten the layer.
- dense layers for classification, with seven output neurons in the last layer

Recommendations

- Collect a larger and more diverse dataset that includes a balanced representation of all emotions in order to improve model generalisation.
- Advanced model architectures: Investigate more sophisticated CNN architectures or transfer learning methods to potentially improve performance.
- Fine-tuning Experiment with various hyperparameters, such as learning rate, batch size, and regularisation techniques, to improve model training.
- Feature Engineering: Investigate additional features, such as facial landmarks or texture analysis, to provide more informative data to the model.
- Real-world Evaluation: Run the model on real-world data to see how it performs in practical scenarios.

Results

The model achieved an overall accuracy of 65% on the test set. While this demonstrates some ability to recognize emotions, there is room for improvement. Notably, the model performed better on recognizing happiness (precision: 75%, recall: 80%) compared to disgust (precision: 50%, recall: 45%), possibly due to a smaller number of disgust images in the dataset.

1.2)Summarization in Videos

Objective

The goal of this project is to condense a video into a manageable chunk while keeping important and distinctive frames. With a duration of roughly 15 seconds, the condensed video is considerably shorter than the original, which was 1 minute and 14 seconds. By examining frame differences and taking periodic frame samples, this is accomplished.

Methodology Used

1. Calculating Frame Difference
   Determine the absolute differences between consecutive frames, pixel by pixel.
   To calculate the average per-pixel difference, normalise the differences using the frame size.
   Utilise a predetermined cutoff point to identify noteworthy alterations.

2. Sampling of frames:
   Determine the total number of frames needed for the shortened video by dividing the desired length by the frame rate of the video.
   To reach the desired frame count, ascertain the sampling interval for frames.
   Periodically sample frames and assess them according to the frame difference.

3. Threshold:
   To determine which frames are significant, compare the computed frame differences to a threshold value.
   If a frame's difference surpasses a certain threshold, signifying a significant change in content, include it in the summarised video.

Libraries and Algorithms:

1.For frame capture and video processing, **use OpenCV**.
2.For effective numerical operations and the computation of frame differences, use **NumPy**.
3.A **time module** for gauging performance.

Recommendations

- Dynamic Thresholding: To enhance frame selection under different video circumstances, use adaptive thresholding.
- Feature-Based Summarization: Identify and extract frames based on particular features or objects of interest (e.g., faces, actions) using machine learning techniques.
- Real-Time Summarization: Create real-time video stream summarization tool

Results:

- The summarized video duration was approximately 15 seconds, significantly shorter than the original video.
- The frame difference threshold effectively filtered out redundant frames, focusing on significant changes.
- The performance metrics indicated efficient processing with a reasonable frames-per-second rate.