

# VIDEO INDEXING

STEP1:

## VIDEO PROCESSING





Purpose of Video Processing



Convert videos  
to a standard  
format (MP4).

Extract frames  
for visual  
analysis.

Identify keyframes  
using visual and  
audio cues for  
efficient processing.



Techniques Used



# VIDEO FORMAT CONVERSION



**TOOL**

FFmpeg (ffmpeg-python).



**HOW**

Converts non-MP4 videos to MP4 using libx264 (video) and aac (audio) codecs.



**WHY**

Standardizes input for consistent processing in later steps.





Techniques Used



# FRAME EXTRACTION



**TOOL**  
OpenCV (cv2).



**HOW**  
Extracts frames at a specified rate (1 FPS), resizes to 640x360, and saves with timestamps in JSON.



**WHY**  
Provides visual data for analysis (e.g., object detection in Step 2).



Techniques Used



# KEYFRAME EXTRACTION



**TOOL**

OpenCV for visual cues.



**HOW**

**Visual Cues:** Uses HSV histogram differences (Bhattacharyya distance) to detect scene changes.  
**Saves keyframes with timestamps in JSON.**



**WHY**

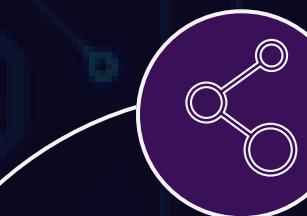
Selects representative frames, reducing data for efficient processing while capturing key moments.



## Importance of Step 1



**ENABLES  
DOWNSTREAM  
STEPS**



**SUPPORTS  
MULTIMODAL  
PROCESSING**



**IMPROVES  
EFFICIENCY**



**ENSURES  
COMPATIBILITY**

Frames and keyframes feed into feature extraction (Step 2) and segmentation (Step 3).

Example: Keyframes with a dog are analyzed for objects or text.

Combines visual (histograms) and audio (energy) cues for robust keyframe detection

Keyframes reduce the number of frames processed, saving time and resources.

MP4 conversion guarantees consistent input for tools like OpenCV and FFmpeg.