GROUP - 3

REAL-TIME BACKGROUND SUBTRACTION SYSTEM FOR VIDEO SURVEILLANCE

Table of contents

Introduction

- Create a real-time algorithm to spot moving objects in videos by subtracting the background.
- Study current methods to identify the best features.
- Make the algorithm handle changes in light and movement in the background.
- Use OpenCV and advanced DIP techniques to clear up the image and reduce noise.
- Enhance background removal techniques for sharper video object detection.

Methodology

Step 1: Initialize Flask and set up camera and classifiers.
Step 2: Function to start camera capture from webcam.
Step 3: Capture a reference frame for change detection.
Step 4: Continuously process video frames to detect movement.
Step 5: Encode video frames as JPEG for streaming.
Step 6: Set up Flask routes for webpage display and video feed.
Step 7: Run the Flask app with live video handling capabilities

IMAGE PROCESSING TECHNIQUES

01 GRAYSCALE CONVERSION

02 HSV CONVERSION

03 ABSOLUTE DIFFERENCE

04 THRESHOLDING

05 BITWISE OR

06 EROSION

07 DILATION

08 FOREGROUND EXTRACTION

09 MASK APPLICATION

Grayscale conversion simplifies images to black-and-white, preserving key details.

HSV conversion transforms RGB images to HSV, simplifying color manipulation.

Thresholding in image processing involves converting grayscale images into binary images.

Erosion and dilation operations in image processing used to respectively shrink and expand regions of interest.

Mask application selectively applies image regions based on a defined mask, allowing focused filtering of specific areas.

MODELS

HAARCASCADE_FRONTALFACE_DEFAULT

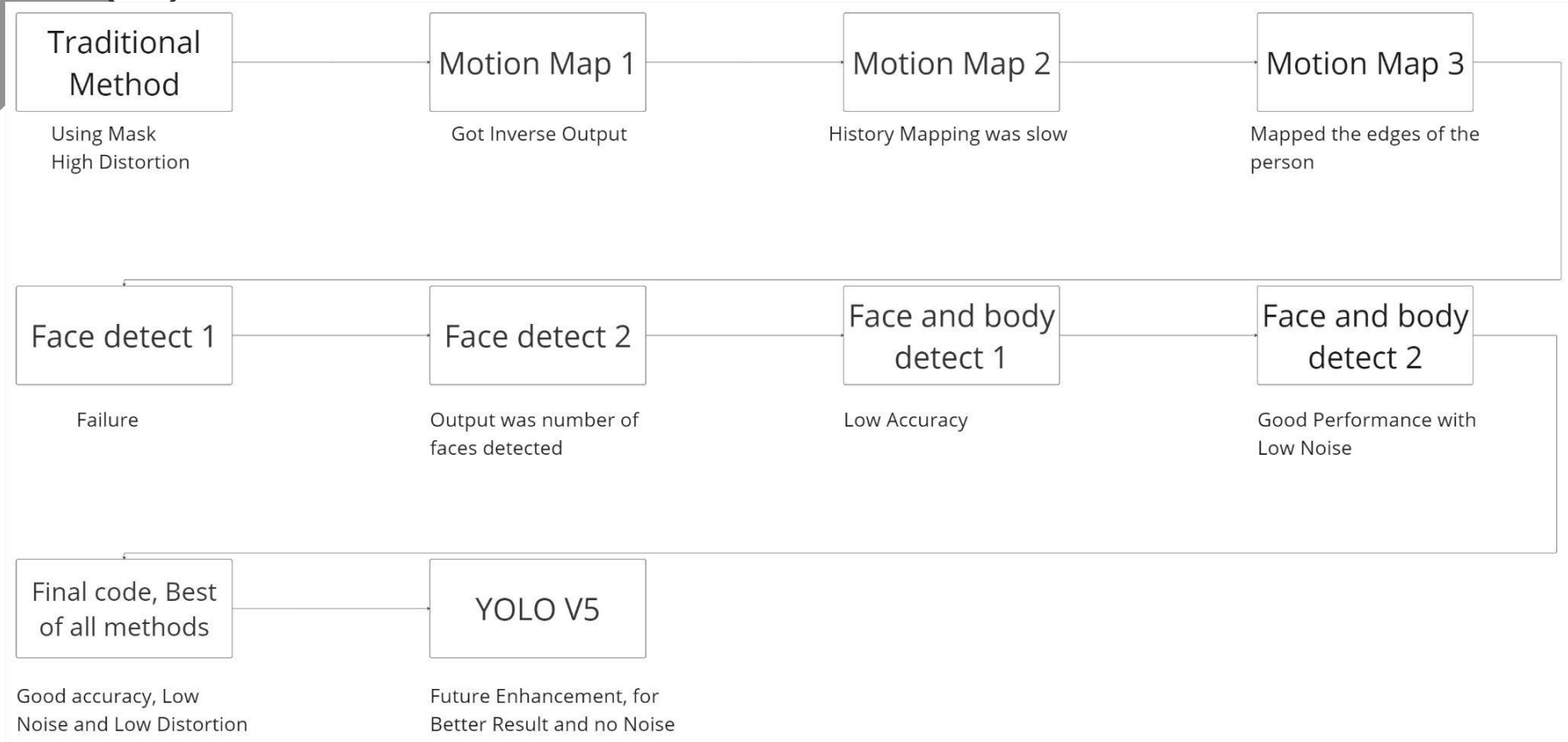
HAARCASCADE_FULLBODY

The haarcascade_frontalface_default model is a pre-trained classifier used for detecting frontal faces in images or video streams.

The haarcascade_fullbody model is a pre-trained classifier designed to detect full-body shapes in images or video streams.

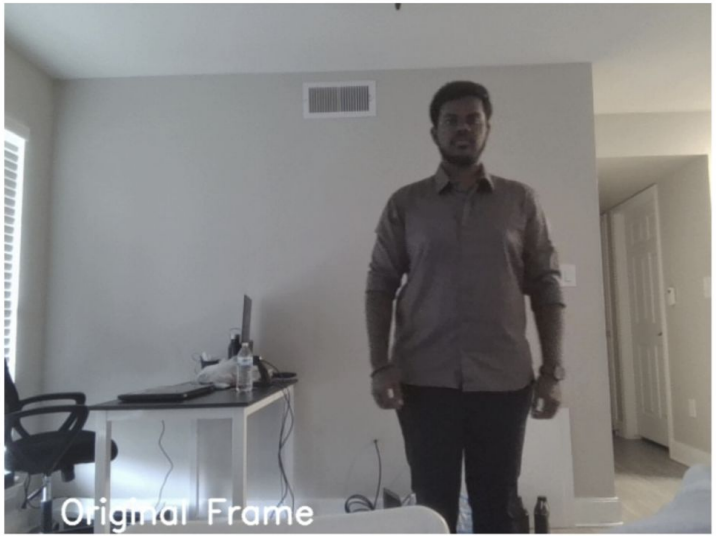
It employs Haar feature-based cascade classifiers, making it efficient for real-time face detection applications.

IMPLEMENTATION PATH



UI AND RESULT

REAL-TIME BACKGROUND SUBTRACTION SYSTEM FOR VIDEO SURVEILLANCE



Conclusion

- Utilized OpenCV for various image processing methods to detect objects.
- Achieved significant results, yet some noise persists in the output.
- For future improvements, consider integrating the YOLO model for enhanced precision.
- Anticipate that YOLO's advanced capabilities will effectively minimize noise and improve detection accuracy.

REFERENCES

- Spagnolo et al. (2006) - Segmented moving objects by background subtraction.
- Minichino & Howse (2015) - Learning computer vision with OpenCV 3 in Python.
- Said & Jambek (2021) - Analyzed image processing with morphological operations.
- Bouwmans et al. (2019) - Reviewed deep learning for background subtraction.
- Lathkar (2021) - Guide on building web apps with Flask framework.