Cluster Innovation Centre
University of Delhi

Forensic Analysis of Image and Video Content

Aryan Sharma (152213) aryang1165@cic.du.ac.in
Mahima Agarwal (152229) agarwalmahima01@cic.du.ac.in
Nitish Singla (152235) nitishsingla159@cic.du.ac.in
Saswat Susmoy Sahoo (152243) saswatsusmoy8013@cic.du.ac.in

Project Mentor: Dr. Manish Kumar

ABSTRACT

In the expansive field of image processing, analysing CCTV footage and video presents a challenging domain that demands substantial knowledge and expertise. Numerous commercially available software solutions are widely employed for forensic applications by diverse law enforcement agencies, often incurring high associated costs. This project seeks to address these challenges by developing an open-source image processing package tailored specifically for forensic applications. The scope encompasses the handling of diverse image and video data types, and manipulating their properties effectively. This project includes fundamental image and video loading, editing with a variety of filters, and offering precise control over various properties.

INTRODUCTION

This project focuses on creating a user-friendly forensic analysis application using Python, integrating OpenCV for video processing, CustomTkinter for GUI development, and VidStab for stabilization. This application also incorporates advanced features like object tracking and frame averaging to enhance video analysis and editing precision. The combination of these technologies ensures a versatile editing environment that adapts to various user needs and improves overall video quality. It aims to provide an intuitive platform that simplifies video editing for both novice and advanced users, enabling the production of professional-quality content without complex or costly software.

OBJECTIVE

The application will integrate intuitive GUI elements using the CustomTkinter library, ensuring a modern and appealing interface that enhances user interaction and experience. Additionally, the project will incorporate advanced video processing techniques such as adaptive thresholding and smoothing using OpenCV, allowing for detailed manipulation and enhancement of video content. By providing detailed documentation and tutorials, the project aims to empower users to fully utilize the application's capabilities, fostering a community of creative individuals who can share tips and improvements. This approach will not only make advanced video editing more accessible but also encourage continuous development and refinement of the application based on user feedback and evolving video editing needs.

METHODOLOGY

1 Planning Phase:

- Analyzed and documented requirements for advanced video and image processing functionalities including grayscale conversion, object tracking, and video stabilization.
- Selected Python for its powerful image and video processing capabilities, utilizing libraries such as OpenCV and VidStab.
- Choose CustomTkinter for GUI development, focusing on user interaction and accessibility.

2 Development Phase:

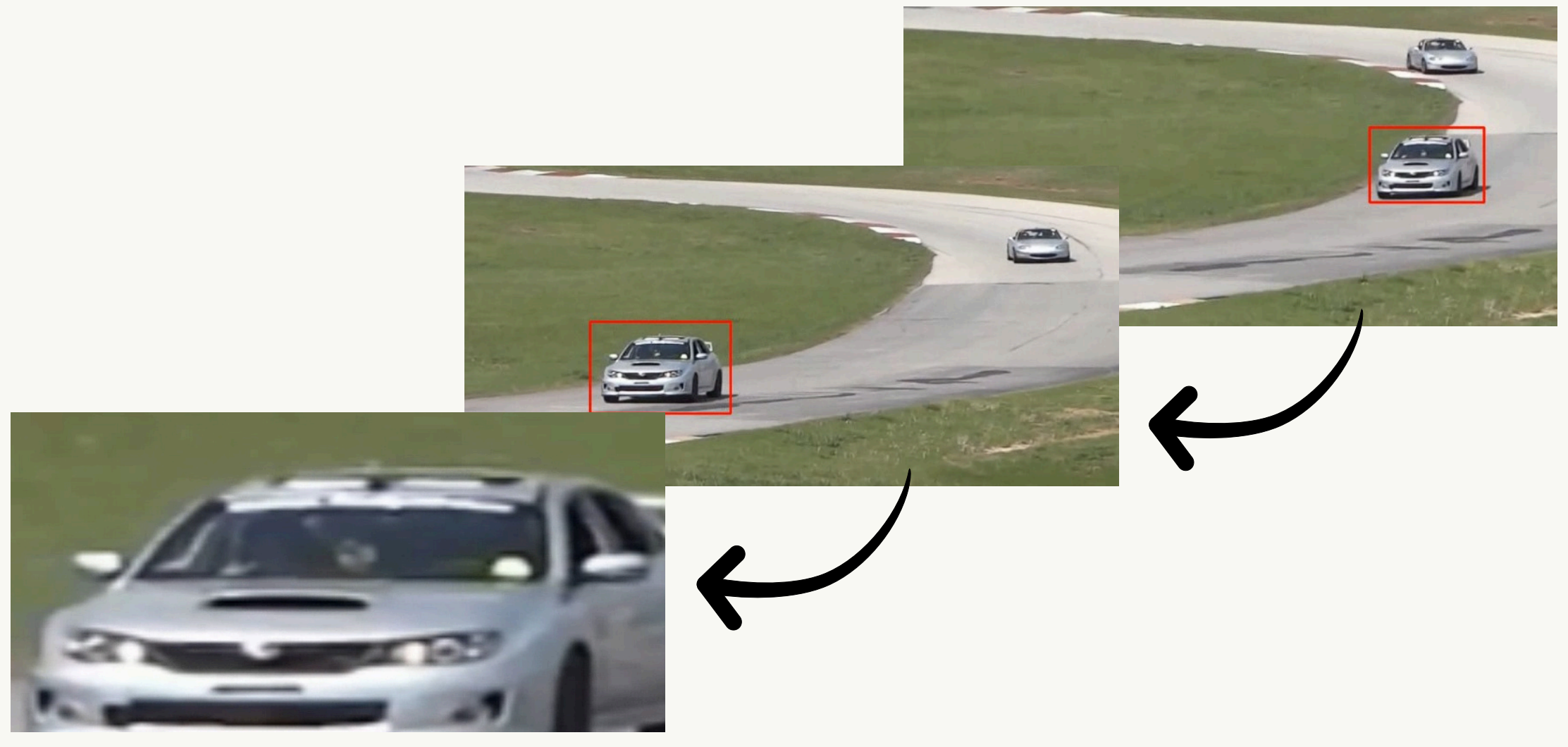
- Implemented various image-processing functions
- Integrated video processing capabilities
- Designed the GUI using CustomTkinter, incorporating:

3 Testing Phase:

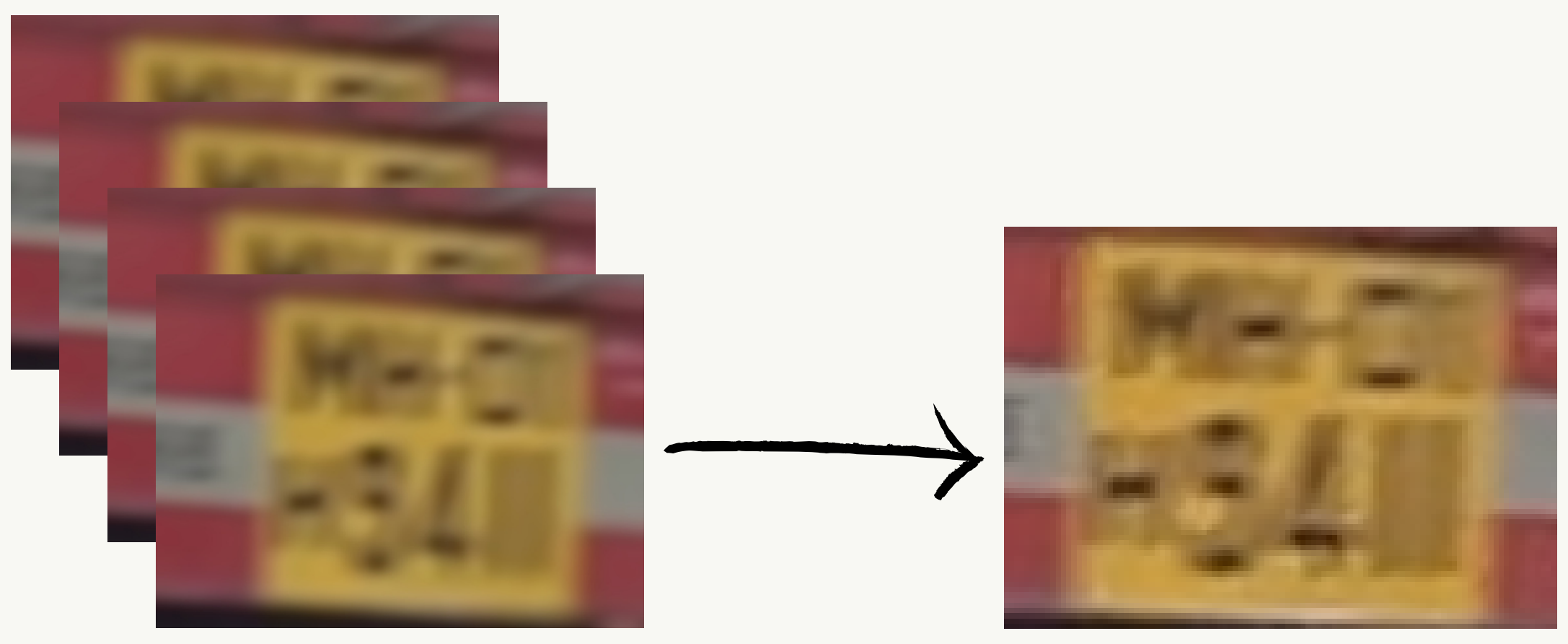
- Conducted rigorous testing to ensure functionality and performance of image and video processing tasks.
- Validated GUI elements for ease of use and effectiveness in user interaction.

RESULTS

The project successfully implemented advanced video and image processing functionalities, including grayscale conversion, object tracking, and video stabilization, using Python with libraries like OpenCV and VidStab.



[Object tracking]



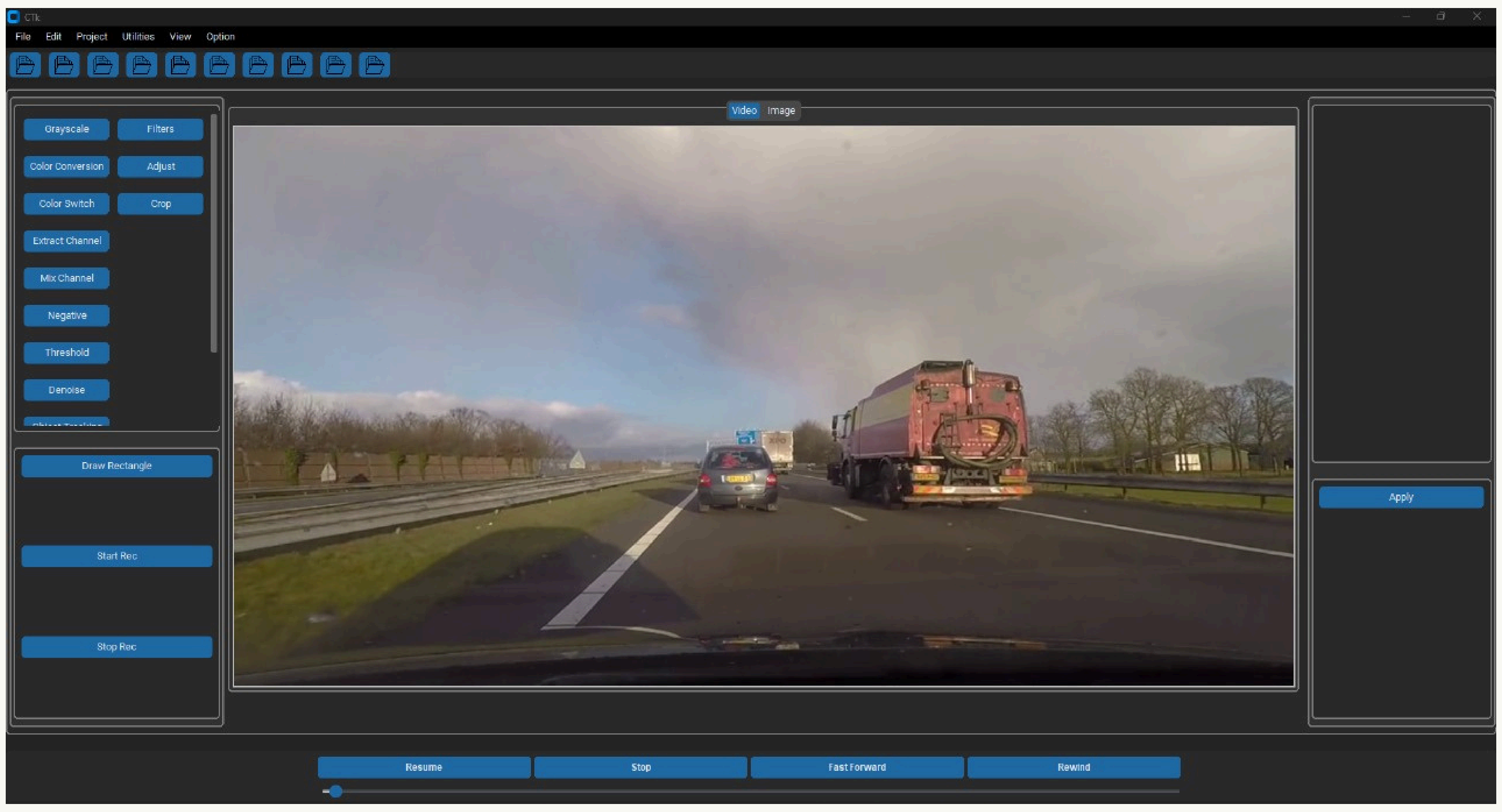
[Original Frames]

[Averaged Frame]

FEATURES

The software provides a comprehensive collection of various video processing features, including object tracking, video stabilization, and frame averaging, using different libraries and methodologies. Object tracking is detailed with multiple algorithms like BOOSTING, MIL, KCF, and others, each with specific attributes and initialization methods. Video stabilization is achieved using the VidStab library, focusing on keypoint detection and trajectory estimation. Frame averaging is explained as a method to enhance video quality by reducing noise, using tools like OpenCV and Matplotlib for implementation.

[Features]



[GUI Window]

FUTURE WORK

The project will focus on optimizing algorithms for enhanced processing of high-resolution videos and complex tasks, and expanding features for real-time video processing and advanced object recognition. User interface improvements will be made based on feedback to increase accessibility and usability. Future versions will integrate with additional third-party APIs and tools, and a mobile version will be developed to extend capabilities to smartphones and tablets. Additionally, the team will collaborate with academic and industry partners to stay at the forefront of image and video processing technology.