Interactive Video Filter App

10th June 2024

Rohan Reddy - PSU ID: 929665094

Abstract

This paper presents the development of an interactive video filter application integrated with real-time object detection capabilities. Utilizing OpenCV for video processing and the YOLO (You Only Look Once) model for object detection, the application offers a user-friendly interface for applying various visual filters, detecting objects, and interacting with video feeds. This system addresses the need for real-time video enhancement and object detection, providing a versatile tool for applications in security, content creation, and interactive media.

Introduction

In the era of digital media, video content has become a ubiquitous part of everyday life. Enhancing videos with visual effects and detecting objects within video feeds are increasingly important for applications in security, entertainment, and data analysis. However, existing solutions often require extensive technical knowledge or do not support real-time processing, limiting their accessibility and usability.

This project aims to create an interactive application that combines real-time video filtering with object detection using machine learning techniques. By leveraging OpenCV for video manipulation and the YOLO model for object detection, the application provides a seamless and efficient way to enhance video content and identify objects in real-time. The user-friendly interface further ensures that users of all skill levels can easily interact with and utilize the application.

Problem Statement

With the growing popularity of video content, there is a rising demand for tools that allow users to enhance their videos with visual effects and perform object detection. Existing solutions often lack real-time processing capabilities or require extensive technical knowledge to operate. This project aims to address these challenges by developing an

application that combines real-time video filtering with object detection, accessible through a user-friendly interface.

Method

The development of the interactive video filter application involved several key components, each contributing to the overall functionality and user experience of the system.

User Interface

The user interface (UI) is designed to be intuitive and easy to navigate. It includes buttons for selecting various filters, a slider for adjusting filter intensity, and options for loading new videos and taking snapshots. The UI is implemented using OpenCV's GUI functionalities, which allow for real-time updates and interactions.

Filter Application

The application supports a range of visual filters, which are applied to the video feeds in real-time. These filters include:

- **Gaussian Blur:** Reduces image noise and detail by averaging pixel values.
- **Median Filter:** Removes noise by taking the median value of neighboring pixels.
- **Bilateral Filter:** Smooths images while preserving edges.
- **Laplacian:** Highlights edges by calculating the second derivative of the image.
- **Color Space Conversions:** Converts the video to different color spaces, such as RGB, HSV, and LAB.

The filters are applied using OpenCV functions, with the intensity of each filter adjustable via a slider in the UI. This allows users to customize the appearance of the video feed to their preferences.

Object Detection

Object detection is implemented using the YOLOv3 model, which is known for its speed and accuracy. The model is loaded and executed using OpenCV's DNN (Deep Neural Network) module. YOLO divides the image into a grid and predicts bounding boxes and

class probabilities for each cell in the grid. The predictions are processed to remove redundant boxes and highlight the most likely detections.

The key steps involved in object detection are:

1. **Loading the Model:** The pre-trained YOLOv3 model and its configuration and class files are loaded into the application.
2. **Preprocessing:** Each video frame is preprocessed into a format suitable for the YOLO model.
3. **Inference:** The preprocessed frames are passed through the model to obtain predictions.
4. **Postprocessing:** The predictions are processed to draw bounding boxes around detected objects and label them appropriately.

Real-Time Processing

To achieve real-time performance, the application continuously reads frames from the video feed, applies the selected filter, performs object detection (if enabled), and displays the processed frames. This is done within a main loop that ensures smooth and efficient updates to the video display.

Experiments

To evaluate the performance and usability of the interactive video filter application, several experiments were conducted:

Experiment 1: Real-Time Filter Application

Objective: Assess the ability of the application to apply various filters in real-time without significant latency.

Method: Different filters were applied to a series of video feeds, and the responsiveness and quality of the filtered video were observed.

Results: The application successfully applied all supported filters in real-time, with minimal latency. The filter intensity slider allowed for smooth adjustments, and the visual quality of the filtered video remained high.

Experiment 2: Object Detection Accuracy

Objective: Evaluate the accuracy and reliability of the YOLOv3 model for real-time object detection.

Method: The application was tested on videos containing various objects from the COCO dataset, which YOLOv3 is trained on. The number of correctly detected objects, false positives, and missed detections were recorded.

Results: The YOLOv3 model demonstrated high accuracy in detecting and labeling objects in real-time. The model effectively highlighted and labeled objects, with a low rate of false positives and missed detections.

Experiment 3: User Interface Usability

Objective: Test the usability and intuitiveness of the user interface.

Method: A group of users with varying technical backgrounds interacted with the application, applying filters

Results and Discussion

The application successfully integrates real-time video filtering and object detection, providing a seamless user experience. Users can apply filters to videos, detect objects, and adjust filter intensity through a simple interface. The object detection feature effectively highlights and labels detected objects, demonstrating the power of the YOLO model in real-time applications.

Future Opportunities

There are several areas for future enhancement:

- **Advanced Object Detection Models:** Incorporate more advanced models like YOLOv4 or Faster R-CNN for improved accuracy and performance.
- **Additional Filters:** Expand the range of available filters and effects.
- **User Customization:** Allow users to upload custom filters and models.
- **Performance Optimization:** Improve the efficiency and speed of the application for smoother real-time processing.
- **Cloud Integration:** Deploy the application on cloud platforms to handle larger videos and more complex processing tasks.

Conclusion

This project demonstrates the successful development of an interactive video filter application with object detection capabilities. By leveraging OpenCV and the YOLO model, the application provides a powerful tool for real-time video enhancement and object detection, all through an intuitive user interface.

References

- OpenCV Library: https://opencv.org/
- YOLO (You Only Look Once) Model: https://pjreddie.com/darknet/yolo/
- COCO Dataset: http://cocodataset.org/