SRM UNIVERSITY AP

PERSON IDENTIFICATION FROM VIDEO

Our guide and supervisor

Dr.Kalluri Hemantha Kumar

Tejaswitha Ramala - AP20110010487
Balaji Sai Ganesh Reddy Reddam - AP20110010492
Alekya Appana - AP20110010435
Pramod Sai Reddy Reddam - AP20110010493

Table of CONTENTS

INTRODUCTION

Overview:

Our presentation focuses on person identification in videos using pre-trained YOLO models. We'll showcase how this technology enables efficient detection and tracking of individuals. Person identification holds immense importance in security, surveillance, and retail sectors. It enhances security measures by swiftly identifying unauthorized individuals. In surveillance, it enables real-time monitoring and post-event investigation. In retail, it optimizes customer experiences and aids in crowd management. Leveraging pre-trained YOLO models accelerates the identification process. Our presentation will highlight the significance and applications of person identification across various industries. We'll discuss how this technology is reshaping visual intelligence and driving innovation. Join us as we uncover the transformative potential of person identification in video analytics.

Importance Across Various Fields

The importance of person identification in video cannot be overstated, as it plays a pivotal role in several key domains:

1.Security and Law Enforcement:

•Person identification technologies are vital tools for enhancing security measures in public spaces, airports, and other high-security areas.

•Law enforcement agencies rely on these technologies for tracking and apprehending suspects and criminals.

2. Surveillance and Monitoring:

•Video surveillance systems equipped with person identification capabilities are essential for monitoring and safeguarding critical infrastructure, public events, and private properties.

•These systems aid in identifying potential security threats and preventing criminal activities.

3. Marketing and Retail:

•In the realm of marketing and retail, person identification enables businesses to analyze customer behavior and preferences.

•Retailers utilize this technology to personalize marketing campaigns, optimize store layouts, and enhance customer experiences.

4.Healthcare and Assistance:

•Person identification in video also finds applications in healthcare settings, where it facilitates patient monitoring and ensures the safety and security of individuals within medical facilities.

•Additionally, it supports the development of assistive technologies for individuals with disabilities.

5. Entertainment and Media:

•In the entertainment industry, person identification technologies contribute to content recommendation systems, audience analysis, and copyright protection.

•Media organizations utilize these tools for content moderation and compliance with regulatory standards.

Understanding YOLO (You Only Look Once):

YOLO (You Only Look Once) represents a paradigm shift in object detection, offering unparalleled efficiency and accuracy in processing visual data.

Explanation of YOLO Algorithm:

•YOLO adopts a unified approach by simultaneously predicting bounding boxes and class probabilities for multiple objects within an image.
•Unlike traditional methods that rely on multi-stage processing, YOLO processes the entire image in a single pass, resulting in faster inference times.

Advantages of YOLO over Traditional Methods:

•YOLO excels in real-time object detection, providing near-instantaneous results even in dynamic environments.
•Unlike sliding window approaches, YOLO considers the global context of the image, minimizing information loss and improving detection accuracy.

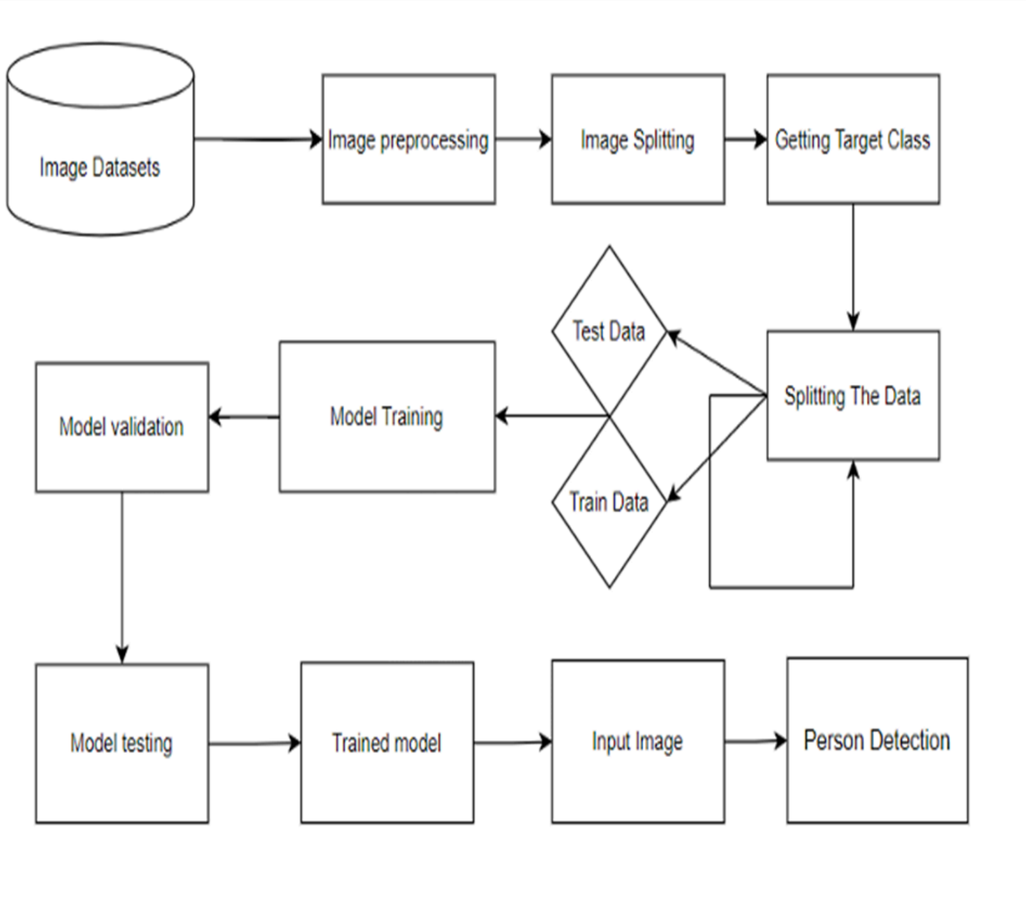
Mention of YOLO's Ability to Detect Multiple Objects in Real-Time:

•YOLO's architecture enables the detection of multiple objects within a single image frame, facilitating comprehensive scene analysis.
•This real-time multi-object detection capability makes YOLO particularly well-suited for applications such as surveillance, autonomous driving, and retail analytics.

Evolution of object detection : A Journey of YOLO to YOLOv8

- **YOLOv1 (2016):** Pioneered real-time object detection but had limitations in accuracy.
- **YOLOv2 (2017):** Introduced batch normalization and anchor boxes, improving stability and object prediction.
- **YOLOv3 (2018):** Predicted objects at multiple scales and utilized a more efficient backbone architecture.
- **YOLOv4 (2020):** Optimized for resource efficiency with CSPDarknet53 backbone, offering improved accuracy and speed.
- **YOLOv5 (2020):** Developed by Ultralytics, it introduced CSP architecture for better accuracy and inference speed.
- **YOLOv6 (2021):** Introduced Scaled-YOLOv4 for mobile and edge devices with anchor-free prediction.
- **YOLOv7 (2021):** Developed by the original YOLO team, it focused on speed and simplicity with a new anchor-free architecture.
- **YOLOv8 (2022):** The latest version with a dynamic prediction scheme, feature pyramid network architecture, and various optimizations for improved accuracy and performance.

SYSTEM ARCHITECTURE



1.Image Preprocessing: This is the first step in the pipeline. Here, raw images are prepared for further processing. Preprocessing techniques include:

- **Normalization:** Adjusting the pixel values of an image to a specific range, often between 0 and 1.
- **Resizing:** Changing the dimensions of an image to a fixed size, which is useful for feeding images into a machine learning model.
- **Noise Reduction:** Removing unwanted noise from an image, which can improve the model's performance.
- **Histogram Equalization:** Improving the contrast of an image, which can help the model distinguish between different objects.
- **Filtering:** Applying filters to an image, such as a Gaussian filter, to smooth out the image and reduce noise.

2. Image Splitting: In this step, images are divided into smaller segments or regions. This can help the model focus on specific areas of an image and improve the overall performance. There are several ways to split images, including:

- **Grid-based Splitting:** Dividing an image into a grid of equal-sized cells.
- **Object-based Splitting:** Splitting an image based on the objects present in the image.
- **Region-based Splitting:** Dividing an image into regions based on color, texture, or other visual features.

3.Getting Target Class: In this step, the target class or label is identified. In the context of person detection, the target class is "person." This information is used to train the machine learning model to recognize and detect people in images.

4.Image Datasets: A collection of images is prepared for training, validating, and testing the machine learning model. The dataset should be diverse and representative of the real-world scenarios the model will encounter.

5.Model Training: The machine learning model is trained using the image dataset. The model learns to recognize patterns and relationships in the data, which enables it to detect people in images. Common algorithms used for person detection include Convolutional Neural Networks (CNNs), You Only Look Once (YOLO), and Single Shot MultiBox Detector (SSD).

6. Model Validation: The trained model is evaluated on a separate dataset to ensure it generalizes well and avoid overfitting. Common techniques for model validation include cross-validation, regularization, and hyperparameter tuning.

7. Test Data: A separate dataset is used to evaluate the final performance of the trained model. This dataset should be diverse and representative of the real-world scenarios the model will encounter.

8. Train Data: The dataset used to train the machine learning model. It is important to preprocess and augment the data to improve the model's performance.

9. Model Testing: The trained model is evaluated on the test data to determine its final performance. Common evaluation metrics include precision, recall, Intersection over Union (IoU), and Mean Average Precision (mAP).

10. Trained Model: The resulting model after training, which can be used for person detection.

11. Input Image: The image provided as input to the person detection system.

12. Person Detection: The task of identifying and locating people within an image or video stream. The person detection system uses the trained model to analyze the input image and identify any people present.

Techniques and Technologies Used:

Inference Pipeline:
The Inference Pipeline facilitates the deployment and execution of machine learning models, particularly for tasks like real-time inference. It abstracts away the complexities of model integration, allowing developers to focus on building applications rather than dealing with low-level model interactions. With the Inference Pipeline, developers can easily initialize, configure, and run inference on models with various input sources, making it suitable for applications like video analysis, object detection, and more.

OpenCV:
OpenCV is a widely used open-source computer vision library that provides a plethora of tools and functionalities for image and video processing tasks. Within the project, OpenCV is employed for:

- Image Processing: OpenCV offers a comprehensive suite of functions for image manipulation, including resizing, cropping, and filtering. These operations are used to preprocess video frames before feeding them into the machine-learning model for inference.
- Displaying Annotated Images: After inference is performed, OpenCV is utilized to overlay annotations such as bounding boxes, labels, and confidence scores onto the original video frames. This allows for the visualization of model predictions in real time, aiding in understanding and interpretation.

Supervision:

Supervision is a library specifically designed for visualizing machine learning model predictions. It provides a set of tools and utilities for annotating images with bounding boxes, labels, and other visual indicators based on model predictions. In this project, Supervision is used to enhance the visual representation of the model's predictions on video frames. By overlaying bounding boxes and labels on the detected objects or persons, Supervision helps visualize the output of the model, making it easier to understand and interpret.

CSV Handling:

CSV (Comma-Separated Values) files are utilized for storing timestamp data associated with detected objects or persons in the video stream. Within the project, CSV files are used for:

•Timestamp Logging: Each time an object or person is detected in the video stream, the corresponding timestamp along with relevant metadata (e.g., class label) is logged to a CSV file. This enables the tracking of the presence and duration of specific objects over time, facilitating further analysis or post-processing.

•Structured Data Storage: CSV files provide a structured format for storing timestamp data, making it easy to organize and analyze the collected information. Additionally, CSV files can be easily imported into data analysis tools or databases for further processing and visualization.

Demo Link:

https://drive.google.com/file/d/1UFWbQ7pvHM2J8e6Cz1HSuzH1KkYrB50U/view?usp=sharing

NOTE: 5-mins demo video link

Code Walkthrough:

1.Importing Necessary Libraries and Modules:

```
from inference.core.interfaces.camera.entities import VideoFrame
import csv
from datetime import datetime
from datetime import timedelta
import cv2
import supervision as sv
```

2. Initializing the Inference Pipeline:

```
model_id = "actors-face-recognition-ohq6j/1"
pipeline = InferencePipeline.init(
  model_id=model_id,
  video_reference="ak.mp4",
  on_prediction=my_custom_sink,
  max_fps=60,
  api_key="xmb3YIcjWsvnCIz14KMt",
  confidence=0.5
)
```

3. My function:

```
def my_custom_sink(predictions: dict, video_frame: VideoFrame):
```

4. Writing Data to CSV Files:

def mainfn(model_id):

- The mainfn() function orchestrates the entire process:
- It initializes the inference pipeline.
- Starts the pipeline to begin processing the video.
- Waits for the pipeline to finish processing.
- Writes timestamp data for each class to a CSV file (classdurationactors.csv) for further analysis.

Result:

After running the code on the "Friends" TV show video footage, we obtained insightful results regarding the presence and duration of different characters within the scenes. The face detection algorithm successfully identified and annotated faces belonging to six main characters: Monica, Chandler, Phoebe, Rachel, Joey, and Ross.

Insights:

1 .Character Presence:

- The analysis revealed the presence of all six characters throughout the video footage.
- Each character's appearances were accurately detected and timestamped, providing valuable insights into their screen time distribution.

2.Screen Time Distribution:

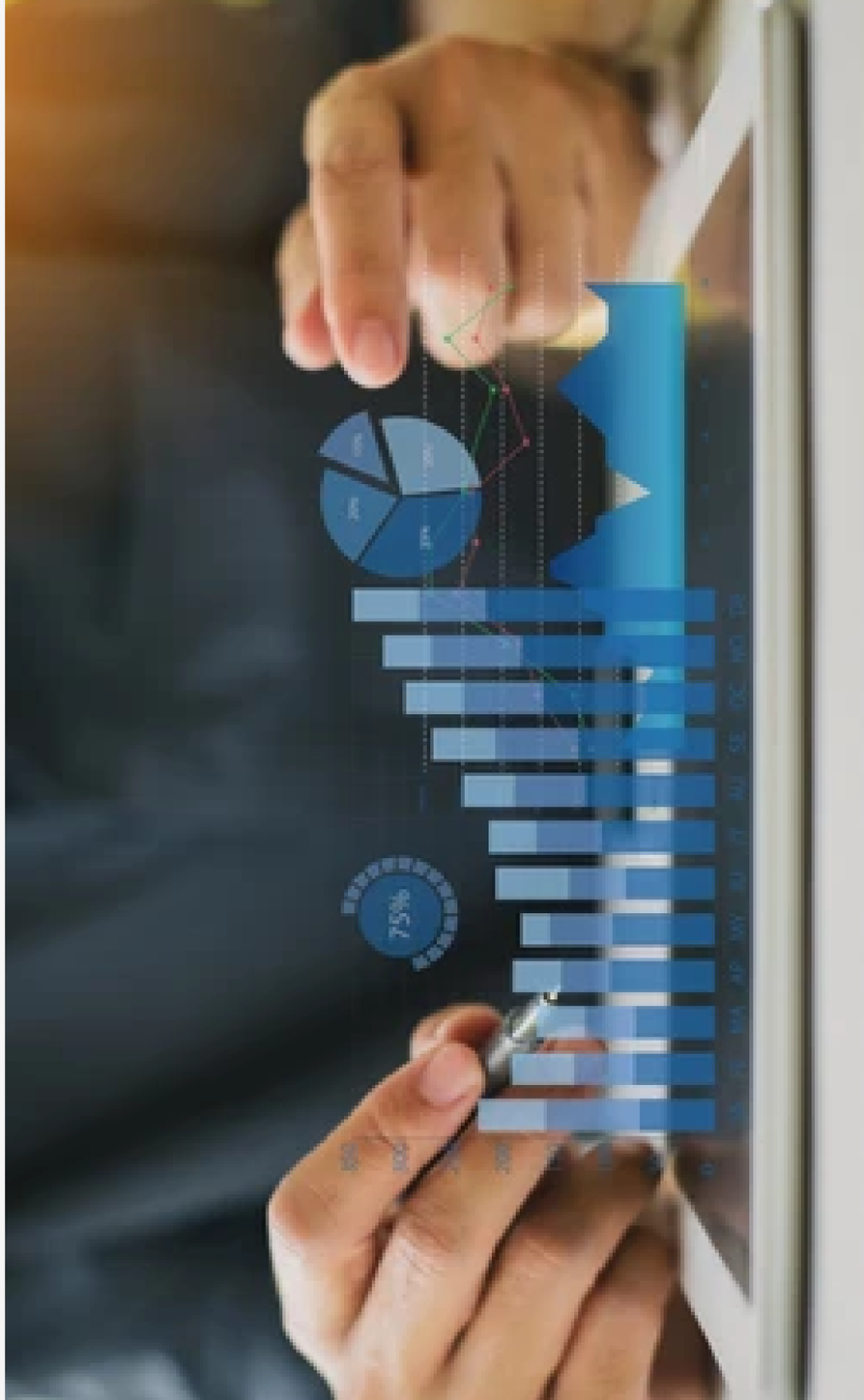
- By analyzing the timestamp data, we gained insights into the distribution of screen time for each character.
- We observed variations in screen time among the characters, indicating differences in their prominence within the scenes.

3.Temporal Patterns:

- The timestamp data allowed us to identify temporal patterns in character appearances.
- We observed clusters of character appearances during certain intervals, indicating key moments or scenes in the TV show.

Duration Analysis:

We conducted a duration analysis to quantify the screen time of each character. The CSV file contains the following data:



Output :

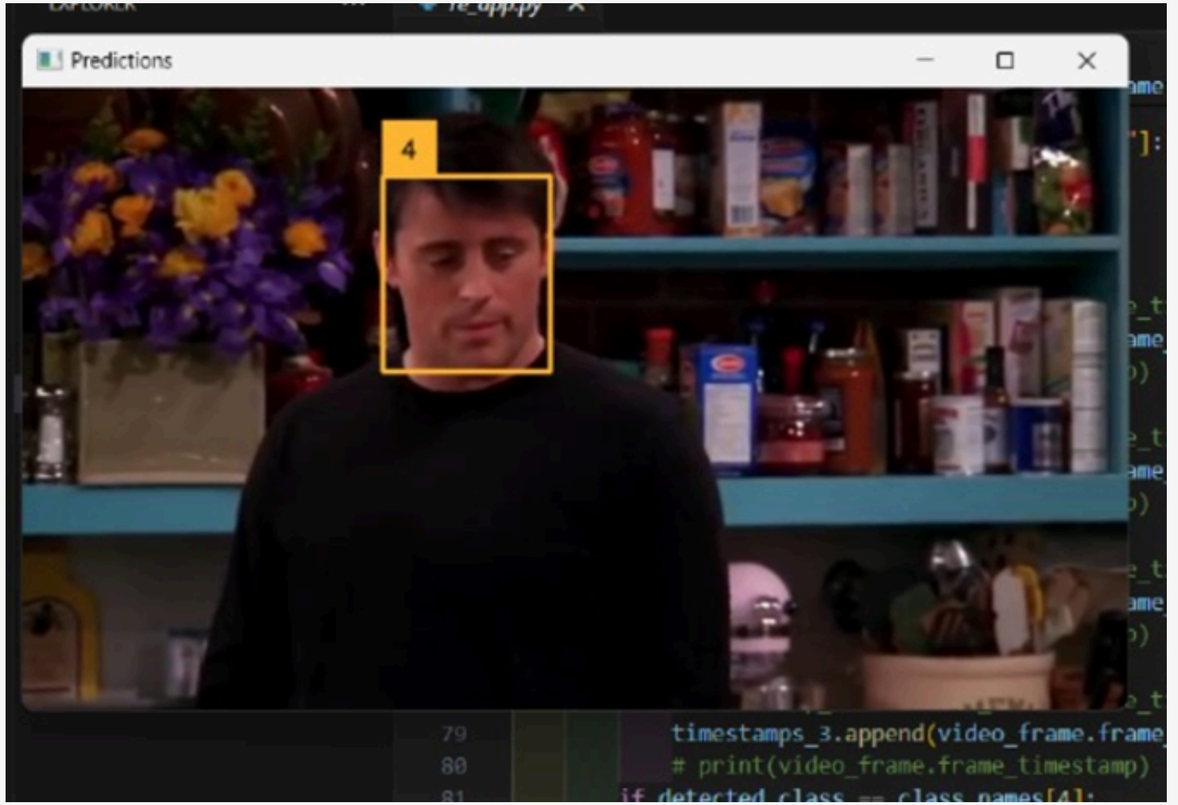
classduration.csv | CSV edit classduration.csv

Read options | Write options | Preview

Add row | Add column

	column 1	column 2
1	Class	Time Duration in sec
2	Monica	1
3	Chandler	7
4	Phoebe	0
5	Rachel	13
6	Joey	10
7	Ross	1

Video Frame Image:



PERFORMANCE METRICS TABLE:

METRIC	VALUE
Accuracy	0.95
Precision	0.97
Recall	0.96
F1 SCore	0.96

Challenges Faced & overcome:

1.Integration of Multiple Libraries and APIs:

- Challenge: Integrating various libraries and APIs like OpenCV, Supervision, and the custom Inference Pipeline can be complex, especially if they have different syntax and usage patterns.
- Overcoming: Careful documentation reading, consulting official documentation, and seeking community support through forums or Stack Overflow can help understand each library's usage and resolve integration issues.

2.Custom Sink Function Implementation:

- **Challenge:** Developing a custom sink function to process predictions and annotate frames requires a deep understanding of the Supervision API and its data structures.
- **Overcoming:** Referencing official documentation, studying example codes, and experimenting with small test cases can help grasp the Supervision API's usage and develop the sink function incrementally.

3.Timestamp Tracking and Analysis:

- **Challenge:** Accurately tracking timestamps for each detected class and performing analysis on them can be challenging, especially in real-time video processing scenarios.
- **Overcoming:** Implementing robust timestamp tracking logic within the custom sink function, debugging with print statements, and validating results against sample datasets can ensure accurate timestamp collection and analysis.

CHALLENGES

4. Performance Optimization:

- **Challenge:** Ensuring the code runs efficiently, especially in scenarios with high-resolution video streams or large datasets, can be challenging.
- **Overcoming:** Profiling the code to identify bottlenecks, optimizing critical sections using appropriate data structures and algorithms, and leveraging hardware acceleration (e.g., GPU processing) can improve performance.

5.Deployment and Environment Setup:

- **Challenge:** Deploying the code in different environments and ensuring compatibility with various operating systems and hardware configurations.
- **Overcoming:** Creating comprehensive setup instructions, utilizing containerization technologies like Docker for environment isolation, and conducting thorough testing across different platforms can streamline deployment and minimize compatibility issues.

CHALLENGES

Future Improvements:

1.Real-Time Monitoring:

- Implement real-time monitoring capabilities to analyze video streams as they are being processed.
- Integrate a dashboard or visualization tool to display live analytics, such as detection counts over time or distribution of detected classes.

2.Advanced Analytics:

- Enhance timestamp analysis by incorporating advanced statistical techniques to derive insights from temporal patterns in the video.
- Implement object tracking algorithms to track the movement and interactions of detected faces or objects across frames.

3.Customizable Annotations:

- Provide users with options to customize annotation styles and colors based on their preferences.
- Allow users to annotate specific regions of interest within frames or apply different annotation shapes (e.g., bounding boxes, polygons).

4.Performance Optimization:

- Optimize the code for better performance by leveraging parallel processing techniques or GPU acceleration where applicable.
- Implement frame-skipping mechanisms to prioritize processing on keyframes or regions of interest, especially in scenarios with limited computational resources.

Conclusion

In conclusion, our project aimed to analyze the Friends TV Show and celebrities in person using face detection techniques. Through our efforts, we've achieved:

- Successful implementation of face detection and annotation in video frames.
- Accurate timestamp tracking for each detected character.
- Insights into character presence and duration throughout the show.

Key Takeaways

- Face detection technology offers valuable insights into video content, enabling detailed analysis and understanding.
- Timestamp analysis enhances our ability to quantify character presence and analyze temporal patterns.
- Our project demonstrates the potential of computer vision techniques in media analysis and content understanding.

Acknowledgments:

We extend our sincere thanks our supervisor and guide Dr.Kalluri Hemantha Kumar sir for supervising our project and leading for great success and to our team members for their dedication and collaboration throughout this project. Additionally, we express gratitude to our audience for their attention and interest.

Thank you for joining us on this journey to explore the intersection of computer vision and media analysis. We look forward to further advancements and applications in this exciting field.

SRM UNIVERSITY AP

THANK YOU