Product Proposal Template

1. Title Page

- Product Name: **Social Distance Detection**
- Proposal Prepared By:
  1. Sai Nikitha Anumakonda (811234655)
  2. Manogna Reddy Voladri (811291665)
  3. Sriya kotu (811294038)
  4. Sai Bhargavi Pusuluri (811284151)
  5. Abhinaya Kapilavai (811289494)
  6. Deepika Mothukuri (811287048)
- Date: 2/12/2024

1. Abstract

- In the situations where social distancing is needed, this tool comes into picture. It is said that a minimum of 6 feet of social distance must be maintained to regulate any spread of viruses. This detection tool is being created to analyze a video feed and notify the users to keep a safe distance from one another. The open-source object detection pre-trained model based on the Yolov3 method was utilized for pedestrian detection, and the video frame captured by the camera served as the input. Using this tool, we can detect people who are not following the social distance, we can warn them and mitigate the risk.

2. Executive Summary

- The Social Distance Detection System is a software solution designed to monitor and enforce social distancing measures in public spaces, workplaces, and other environments. Its purpose is to mitigate the spread of infectious diseases, such as COVID-19, by alerting individuals when they are too close to others. The system aims to deliver enhanced public health and safety by promoting adherence to social distancing guidelines.

3. Product Vision

- **Purpose of the Product**: The Social Distance Detection System utilizes deep learning algorithms to detect the distance between individuals in real-time video feeds. Its intended use is to provide automated monitoring and enforcement of social distancing measures.
- **Target Audience**: The primary users of the product include businesses, organizations, and government agencies responsible for managing public spaces and ensuring compliance with social distancing regulations. Individuals concerned about their health and safety may also benefit from the system.

- **Long-term Vision**: In the future, the social distance detection product is envisioned to become an integral component of public health infrastructure worldwide. As technology continues to advance, the system will evolve to offer enhanced features and capabilities. This may include integration with IoT devices for automated alerts and notifications, and seamless integration with existing security and surveillance systems.

4. Product Value

- **Benefits**:
  - **Enhanced Public Health and Safety:** The primary benefit of the social distance detection product is the promotion of public health and safety by ensuring compliance with social distancing guidelines. By alerting individuals and authorities when safe distances are not maintained, the product helps reduce the risk of spreading contagious diseases like COVID-19.
  - **Risk Mitigation:** Businesses, event organizers, and public institutions can mitigate the risk of outbreaks and potential liabilities by implementing proactive measures to enforce social distancing. This can safeguard their reputation and operations while fostering a safer environment for customers, employees, and visitors.
  - **Operational Efficiency:** Automating the monitoring of social distancing reduces the need for manual supervision, saving time and resources for staff. This allows organizations to focus on other essential tasks while maintaining compliance with health regulations.
  - **Data Insights:** The product generates valuable data insights into crowd behavior, spatial dynamics, and compliance trends. These insights can inform decision-making processes, resource allocation, and future planning to optimize operations and improve overall efficiency.
- **Cost Analysis**: The estimated costs to develop and maintain the product include:
  - Development costs for software engineering and algorithm development.
  - Hardware costs for cameras, sensors, and computing infrastructure.
  - Ongoing maintenance and support expenses.
- **Value Proposition**:
  - **Risk Reduction:** Avoids potential losses from outbreaks and legal issues.
  - **Efficiency Gains:** Improves productivity and resource optimization.
  - **Data-Driven Decisions:** Informs strategic planning and enhances effectiveness.
  - **Public Confidence:** Builds trust and loyalty, enhancing reputation and customer satisfaction.
  - **Potential ROI:** Avoidance of losses and efficiency gains lead to positive returns.

5. Product Creation Outline

- **Design Overview**:
  - YOLOv3 [13] algorithm was used to detect the pedestrian in the video frame.

  - OpenCV for Camera view calibration
  - Finding the distance between two persons using the distance formula.
  - COCO data set for object detection.
  - Video frame for taking the inputs and giving the output.
- **Development Plan**:
  - **Requirement Analysis:**
    - Define specifications and functionalities.
    - Identify key stakeholders and their requirements.
  - **Research and Feasibility Study:**
    - Evaluate existing object detection algorithms.
    - Assess the feasibility of camera calibration techniques.
  - **Prototype Development:**
    - Implement pedestrian detection using YOLOv3.
    - Develop camera view calibration module.
  - **Distance Measurement Algorithm:**
    - Design algorithms to measure distances between pedestrians.
    - Implement scaling factor estimation from camera calibration.
  - **Visualization and User Interface:**
    - Develop a user-friendly interface for input and output visualization.
    - Implement color-coded indicators for safe and unsafe distances.
  - **Testing and Validation:**
    - Test the tool with various video datasets.
    - Validate distance measurement accuracy against ground truth data.
  - **Optimization and Deployment:**
    - Optimize algorithms for efficiency.
    - Prepare the tool for deployment in real-world scenarios.
- **Development Methodology**:
  Agile methodology with a focus on continuous integration and deployment, ensuring that new code changes are thoroughly tested and deployed to production environments quickly and efficiently.
- **Resource Requirements**:
  - Personnel skilled in Python programming, deep learning, computer vision, and software development.
  - Hardware resources including sufficient RAM, processors, and storage capacity.
  - Access to necessary datasets and algorithms.

6. Quality and Evaluation

- **Quality Standards**:
  The Social Distance Detection System will adhere to industry-standard quality benchmarks and best practices. These standards include accuracy in distance measurement, robustness in real-time detection, and user-friendly interface design.
- **Testing Procedures**:
  Rigorous testing, including unit tests, integration tests, and real-world validation, will ensure the system's efficacy.
- **Evaluation Metrics**:
  The success of the Social Distance Detection System will be measured against objectives such as:
  - Accuracy of distance measurement.
  - Detection speed and efficiency.
  - User satisfaction and adoption rates.
  - Reduction in the spread of infectious diseases in monitored environments.

7. Deployment Plans

- **Production Timeline**:
  1. Project Planning : 02/09/2024
  2. Documentation and Pre-requisite data collection : 02/16/2024
  3. Building algorithm : 02/23/2024
  4. To test with the weights and with data set : 03/22/2024
  5. Testing the working of the model : 03/29/2024
  6. Deployment : 04/05/2024
- **Marketing and Distribution Strategy**: The model can be used in offices, public transport, can further be developed to use with cctv photages.
- **Risk Management**: Identify potential deployment risks and mitigation strategies.

8. Maintenance Plans

- **Defects:**
  - **False Positives/Negatives** : Models may produce false positives (identifying social distancing violations where there are none) or false negatives (missing actual violations), leading to inefficiencies or safety risks.
  - **Resource Intensive** : Training and deploying deep learning models for social distancing detection can be resource-intensive, requiring powerful hardware and significant computational resources.
- **Evolution:**

- ○ **Real-time Monitoring** : There's a push towards developing models capable of real-time monitoring to provide instant feedback and intervention in crowded spaces.
- ○ **Integration with Public Health Systems** : Future models may integrate with public health systems to provide insights into social distancing compliance at a community or city level, aiding in policymaking and resource allocation.

9. Literature/product review

- 1.Landing AI Creates an AI Tool to Help Customers Monitor Social Distancing in the Workplace [Onlive]. Available at https://landing.ai/landing-ai-creates-an-ai-tool-to-help-customers- monitor-social-distancing- in-the-workplace/
  (Access on 4 May 2020).
  The COVID-19 pandemic has prompted Landing AI to introduce an AI tool to monitor workplace social distancing adherence, highlighting the importance of compliance in maintaining workplace safety.
  Research on workplace social distancing monitoring emphasizes strategies for compliance, mitigating COVID-19 spread, including clear communication, organizational support, and technological solutions. AI and computer vision technologies offer real-time monitoring of social distancing, enabling proactive intervention and risk mitigation through accurate spatial interaction detection and analysis. Social distancing monitoring tools can improve workplace safety, but concerns about privacy, trust, and psychological well-being need to be addressed through transparent communication, stakeholder engagement, and ethical considerations. Landing AI's AI tool enhances workplace safety by real-time social distancing compliance, but further research is needed to evaluate its effectiveness, scalability, and long-term impact on employee behavior.
  The literature emphasizes the use of technological innovations, particularly AI tools, for monitoring workplace social distancing, but privacy, trust, and ethical concerns must be considered. Interdisciplinary research is crucial for effective strategies.
- 2.R.Girshick,J.Donahue,T.Darrell,J.Malik."Richfeaturehierarchies for accurate object detection and semantic segmentation." In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 580-587. 2014.
  https://arxiv.org/abs/1311.2524

The 2014 paper "Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation" by Girshick, Donahue, Darrell, and Malik revolutionized object detection and semantic segmentation using the Region-based Convolutional Neural Network framework.
Object detection faces challenges due to variations in appearance, scale, and occlusion, with traditional methods being computationally expensive and lacked scalability. The RCNN framework, utilizing deep learning and convolutional neural networks, effectively extracts hierarchical features from images through region proposal generation, feature extraction, and object classification. The

paper introduces the R-CNN framework, enhancing object detection performance on benchmarks like PASCAL VOC and MS COCO, explores transfer learning, and integrates region proposal methods for reduced computational burden. The paper significantly influenced computer vision, leading to advancements in object detection and semantic segmentation. Future research focuses on improving efficiency, exploring multi-task learning approaches, and exploring novel architectures to address limitations of the original R-CNN framework. The paper significantly influenced computer vision, leading to advancements in object detection and semantic segmentation. Future research focuses on improving efficiency, exploring multitask learning approaches, and exploring novel architectures to address limitations of the original R-CNN framework.
The "Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation" revolutionized computer vision by utilizing deep learning and hierarchical features for efficient and scalable object detection.

- 3.J. Redmon, S. Divvala, R. Girshick, A. Farhadi, “You only look once: Unified, real-time object detection”, In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 779-788. 2016.
  https://ieeexplore.ieee.org/document/7780460

The 2016 paper "You Only Look Once: Unified, Real-Time Object Detection" by Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi introduced the YOLO framework, revolutionizing real-time object detection tasks. This literature review examines the paper's impact, contributions, key findings, and subsequent research inspired by its insights.
Object detection is a crucial task in computer vision, with applications in surveillance, autonomous driving, and image understanding. Traditional methods, which involve complex pipelines, lead to computational inefficiencies and reduced speed. The paper introduces the YOLO framework, a unified, end-to-end approach for object detection, predicting bounding boxes and class probabilities directly from a single pass through the neural network, overcoming challenges in separate region proposal and classification stages. The paper introduces the YOLO framework, which outperforms standard object detection benchmarks in accuracy and speed. It showcases real-time object detection capabilities with inference speeds exceeding 45 frames per second on a GPU. The paper also explores grid-based prediction mechanisms, where an image is divided into grid cells, predicting bounding boxes and class probabilities for objects. The paper significantly influenced computer vision, leading to a surge in research on real-time object detection and efficient neural network architectures. Subsequent studies improved the accuracy and efficiency of YOLO-based frameworks through architectural modifications, such as YOLOv2, YOLOv3, and YOLOv4, and adapted the framework for specialized domains like small object detection, instance segmentation, and video object detection. Novel loss functions and training strategies were explored to address class imbalance, localization accuracy, and generalization to unseen object categories.
"You Only Look Once: Unified, Real-Time Object Detection" introduced a unified framework prioritizing

accuracy and speed in object detection. Its innovative grid-based prediction and end-to-end learning continue to inspire research for real-time, efficient, and accurate detection systems.

- 4.K. Simonyan, A. Zisserman, “Very deep convolutional networks for large-scale image recognition”, arXiv preprint arXiv:1409.1556, 2014.
  https://arxiv.org/abs/1409.1556

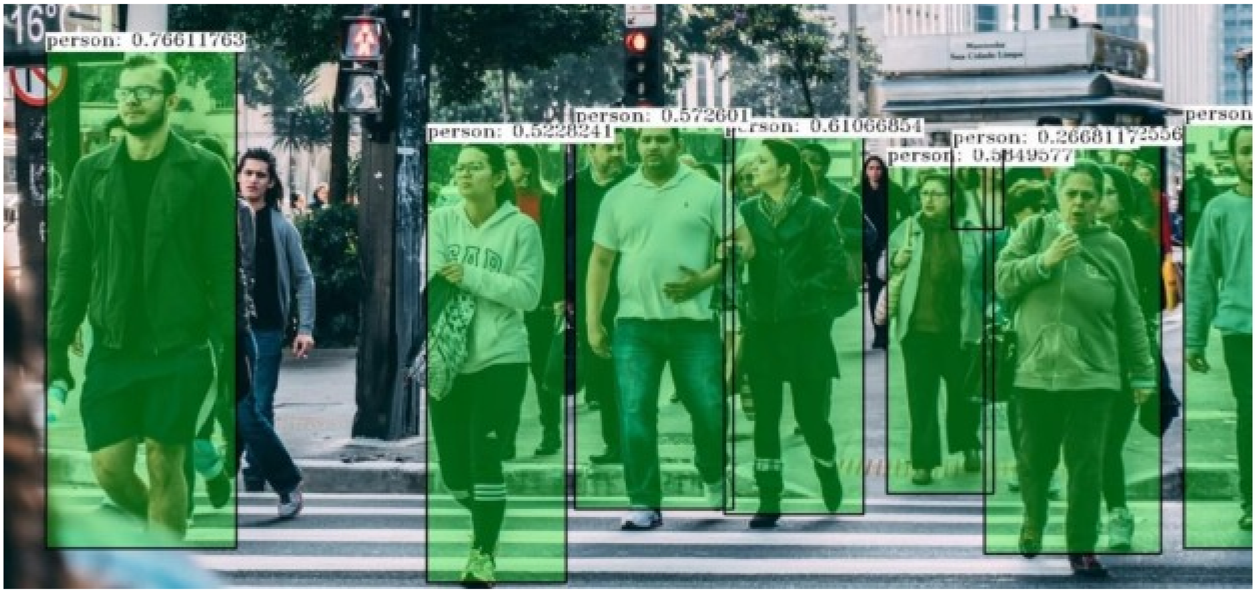
The 2015 paper by Simonyan and Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," introduced the VGG network architecture, significantly advancing image recognition tasks.The VGG paper aims to explore the effectiveness of deep convolutional neural networks (CNNs) for large-scale image recognition tasks, addressing the limitations of traditional shallow architectures and handcrafted features in learning complex hierarchical representations from raw pixel data. The paper proposes a VGG architecture consisting of multiple convolutional layers and max-pooling layers, with deeper stacks compared to previous architectures. Key features include small 3x3 convolutional filters, standardization of network configurations with 16 or 19 weight layers, and the use of max-pooling layers to down sample feature maps and increase translation invariance. The VGG paper highlights the significance of depth in CNN architectures for advanced image recognition performance. It establishes a standardized architecture for deep CNNs, promoting reproducibility and benchmarking. The paper also evaluates the effectiveness of smaller convolutional filters and deeper network architectures in learning hierarchical features from raw image data. The VGG network is introduced as a powerful feature extractor for transfer learning. The VGG paper significantly influenced computer vision research, leading to further studies on deep learning architectures like ResNet, DenseNet, and Inception networks, improving efficiency in deep CNNs, and applying transfer learning and pre-trained VGG models for tasks beyond image recognition.
"Very Deep Convolutional Networks for Large-Scale Image Recognition" significantly advanced deep learning and established deep CNNs as the foundation of modern computer vision systems. Its standardized architecture, deep network configurations, and hierarchical feature learning continue to shape research and development efforts for complex image understanding tasks.

10. Proposed Methods/Technology Demonstration

1. **The use of Technology:**

OpenCV:

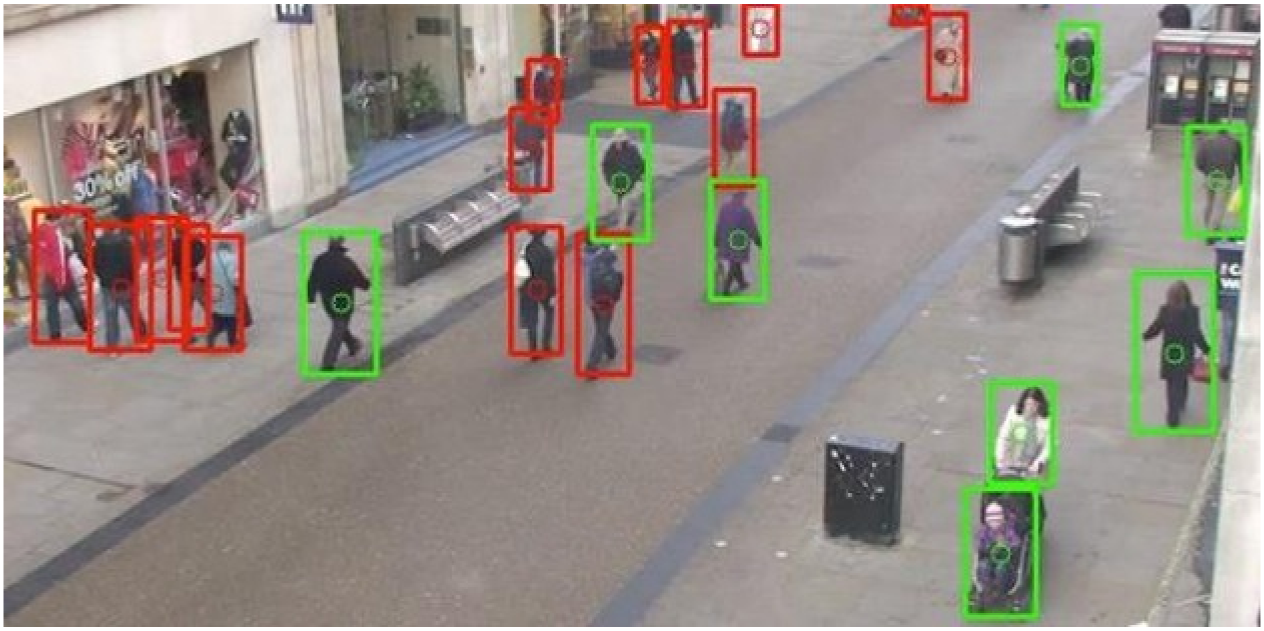
OpenCV, or Open-Source Computer Vision Library, is a popular open-source computer vision and machine learning software library. It provides various tools and functions for image and video analysis, including object detection, facial recognition, motion tracking, and more. There are vast range of optimized algorithms which include computer vision and machine learning algorithms. These algorithms can be used to identify objects, faces, people, classify human actions in videos, classify actions through camera. In our project, OpenCV is being used to detect and identify humans. As we see in the below image, by using OpenCV, we can identify our target object which is person by training it with required dataset.



YOLOV3:

You Only Look Once is a popular and useful algorithm, widely used in the field of object detection. It is known for its speed and accuracy. There are many versions of this algorithm available. After going through several research papers and articles, we have decided to use YOLOV3 in our project as it will provide us our required accuracy and will satisfy the distance specifications.

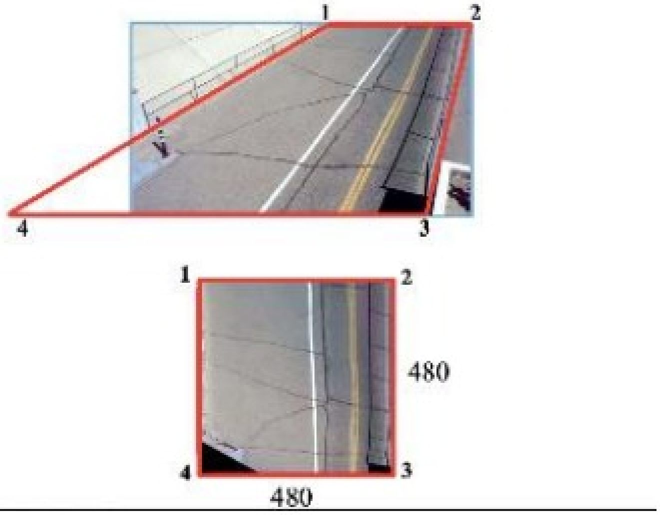
- Initially, we download YOLOv3 and the required configurations. They are available on their official website.
- After which, we load the weights and configurations using OpenCV.
- Then the input given by the user will be converted into a suitable format for feeding into the YOLOV3 model.
- Then it will detect the objects (in our case for human) in the pre-processed image/video through the model. We can see bounded boxes around the objects detected along with the accuracy.
- As we can see in the below image, red boxes indicate that people are not following a safe distance and boxes in green indicate that people obeying the social distance protocol. We can achieve this by using camera view calibration distance calculations.



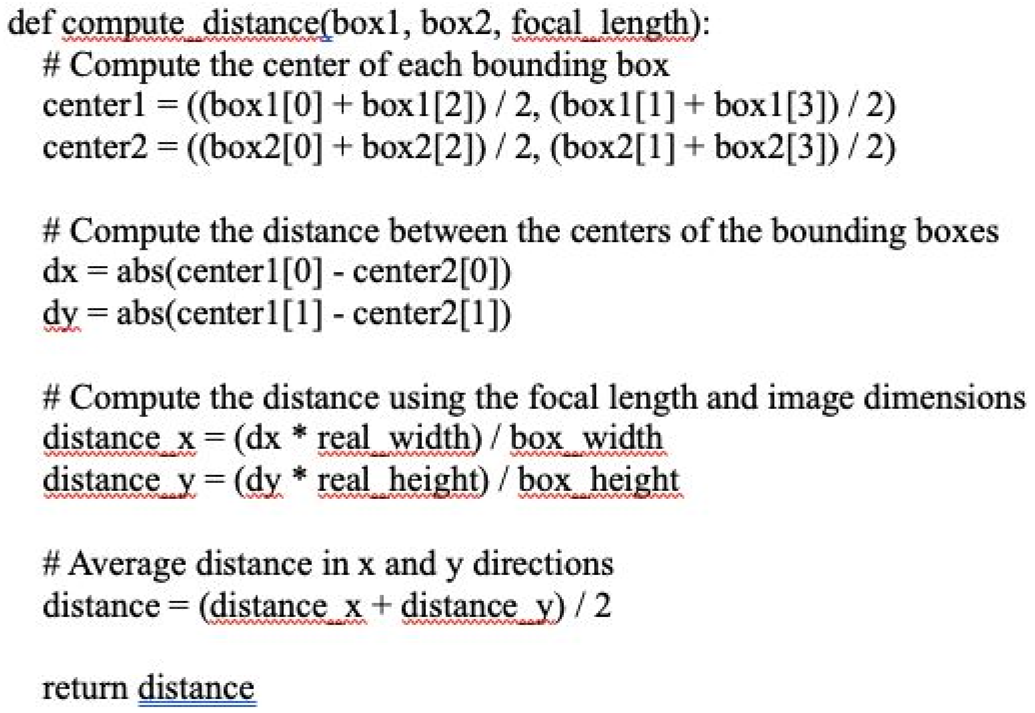
Camera view calibration:

Camera calibration is the process of determining certain parameters of a camera to fulfill desired tasks with specified performance measures. The reader is referred to entry Calibration for a general discussion on calibration. We typically follow the below steps in our project.

- Camera Calibration: Use OpenCV to calibrate your camera using a calibration pattern.
- Object Detection: Use YOLOv3 or another object detection algorithm to detect objects in the scene.
- Estimate Object Dimensions: If you know the real-world dimensions of the objects, you can use these directly. Otherwise, you can use reference objects of known size to calibrate the scale.
- Compute Distance: Once you have the dimensions of the objects in the image, you can use techniques like triangulation or depth estimation to compute the distance between them.



Below is a snippet of code being used in our project to calculate the distance between two people:



```
def compute_distance(box1, box2, focal_length):
    # Compute the center of each bounding box
    center1 = ((box1[0] + box1[2]) / 2, (box1[1] + box1[3]) / 2)
    center2 = ((box2[0] + box2[2]) / 2, (box2[1] + box2[3]) / 2)

    # Compute the distance between the centers of the bounding boxes
    dx = abs(center1[0] - center2[0])
    dy = abs(center1[1] - center2[1])

    # Compute the distance using the focal length and image dimensions
    distance_x = (dx * real_width) / box_width
    distance_y = (dy * real_height) / box_height

    # Average distance in x and y directions
    distance = (distance_x + distance_y) / 2

    return distance
```

Web-based application:

Flask is a micro web framework written in Python. It is classified as a microframework because it does not require particular tools or libraries. It has no database abstraction layer, form validation, or any other components where pre-existing third-party libraries provide common functions.

2. The use of Technology:

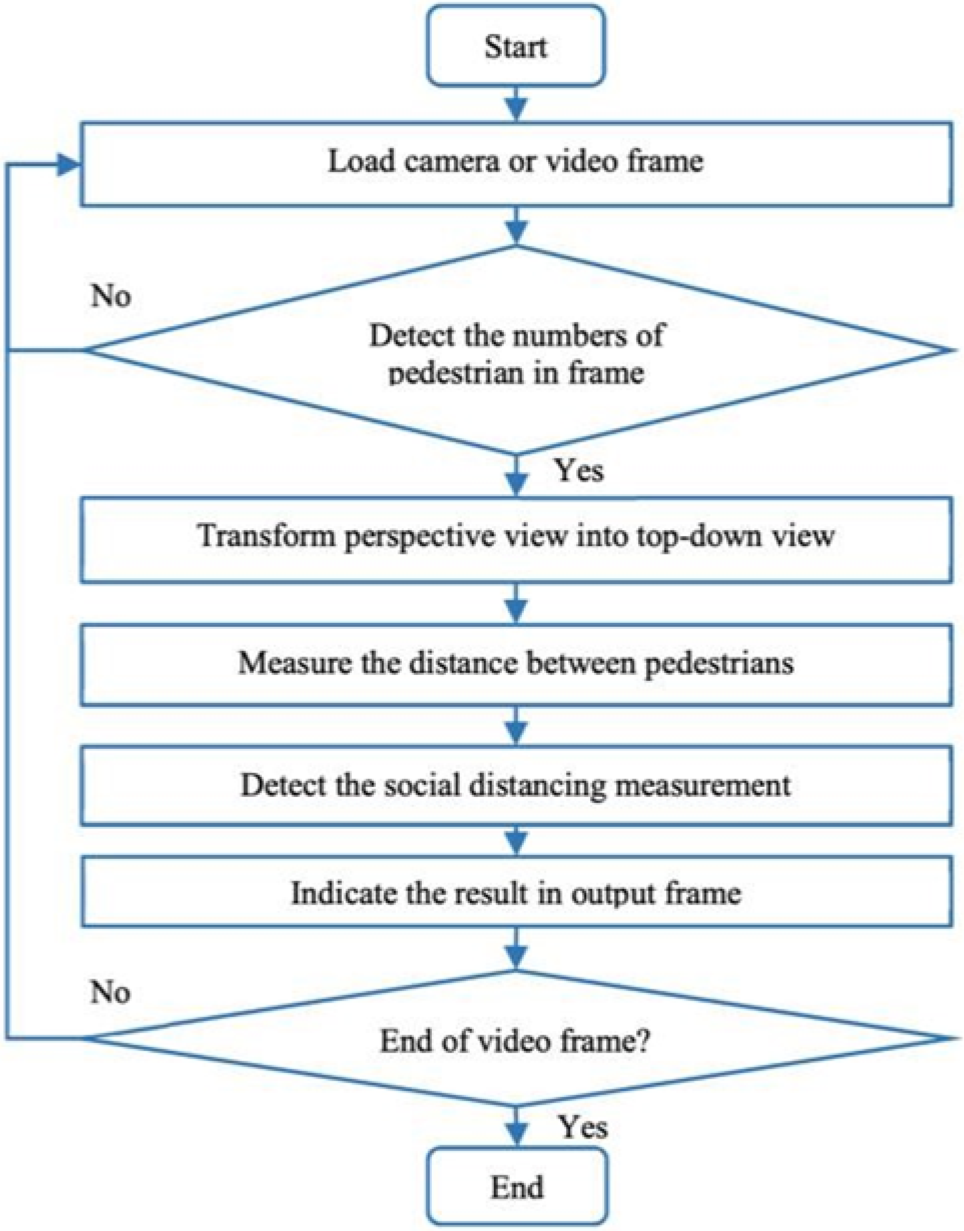
In this repository, we will download the weights required for the project.

https://github.com/patrick013/Object-Detection---Yolov3/blob/master/model/yolov3.weights

Camera View Calibration.

https://www.analyticsvidhya.com/blog/2021/10/a-comprehensive-guide-for-camera-calibration-in-computer-vision/

3.Discussions of how we plan to use the technology


Start
Load camera or video frame
No
Detect the numbers of pedestrian in frame
Yes
Transform perspective view into top-down view
Measure the distance between pedestrians
Detect the social distancing measurement
Indicate the result in output frame
No
End of video frame?
Yes
End

As we see in the above flow chart, there are several steps to be followed. We have had the below discussions on how to implement the project.

• Data collection: We will gather few video footages containing.

• Training Object Detection Models: We will train YOLOv3 models containing videos with people. Also, we will fine tune the models to improve accuracy.

• Camera Calibration: We must capture camera calibration images of known calibration, let's say a chessboard. From different viewpoints. Also detecting calibration pattern corners in the images. Compute camera intrinsic parameters (focal length, principal point) and distortion coefficients using OpenCV's camera calibration functions.

• Integration of Object Detection and Calibration: Develop a unified pipeline that captures video frames from the camera, performs object detection (people) using the trained YOLOv3 models, and applies camera calibration to undistort the frames. After which we use the calibrated camera parameters to

transform the detected bounding boxes from image coordinates to real-world coordinates.
• Social Distance Detection Algorithm: Implement an algorithm to detect social distance violations based on the positions of detected people. We have defined the threshold distance for safe distances as 6 feet for now and use the calibrated real-world coordinates to compute distances between people.
• User Interface Development: Design and develop a user interface to display the video feed, detection results (people, social distance violations), and calibration information.
• Testing and Evaluation: After integrating the framework, we will test its working.

11. Conclusion

The Social Distance Detection System offers a comprehensive solution for monitoring and enforcing social distancing measures in various environments. With adherence to quality standards, robust testing procedures, and a well-defined deployment and maintenance strategy, the product is poised to deliver significant value in promoting public health and safety.

12. Appendices

- Any supporting documents, such as detailed technical specifications, market research data, etc.

13. References

- [1] K. He, X. Zhang, S. Ren, J. Sun, “Deep residual learning for image recognition”, In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 770-778, 2016. https://ieeexplore.ieee.org/document/7780459
- [2] J. Redmon, S. Divvala, R. Girshick, A. Farhadi, “You only look once: Unified, real-time object detection”, In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 779-788. 2016. https://ieeexplore.ieee.org/document/7780460
- [3] J. Redmon, A. Farhadi, “Yolov3: An incremental improvement”, arXiv preprint arXiv:1804.02767, 2018