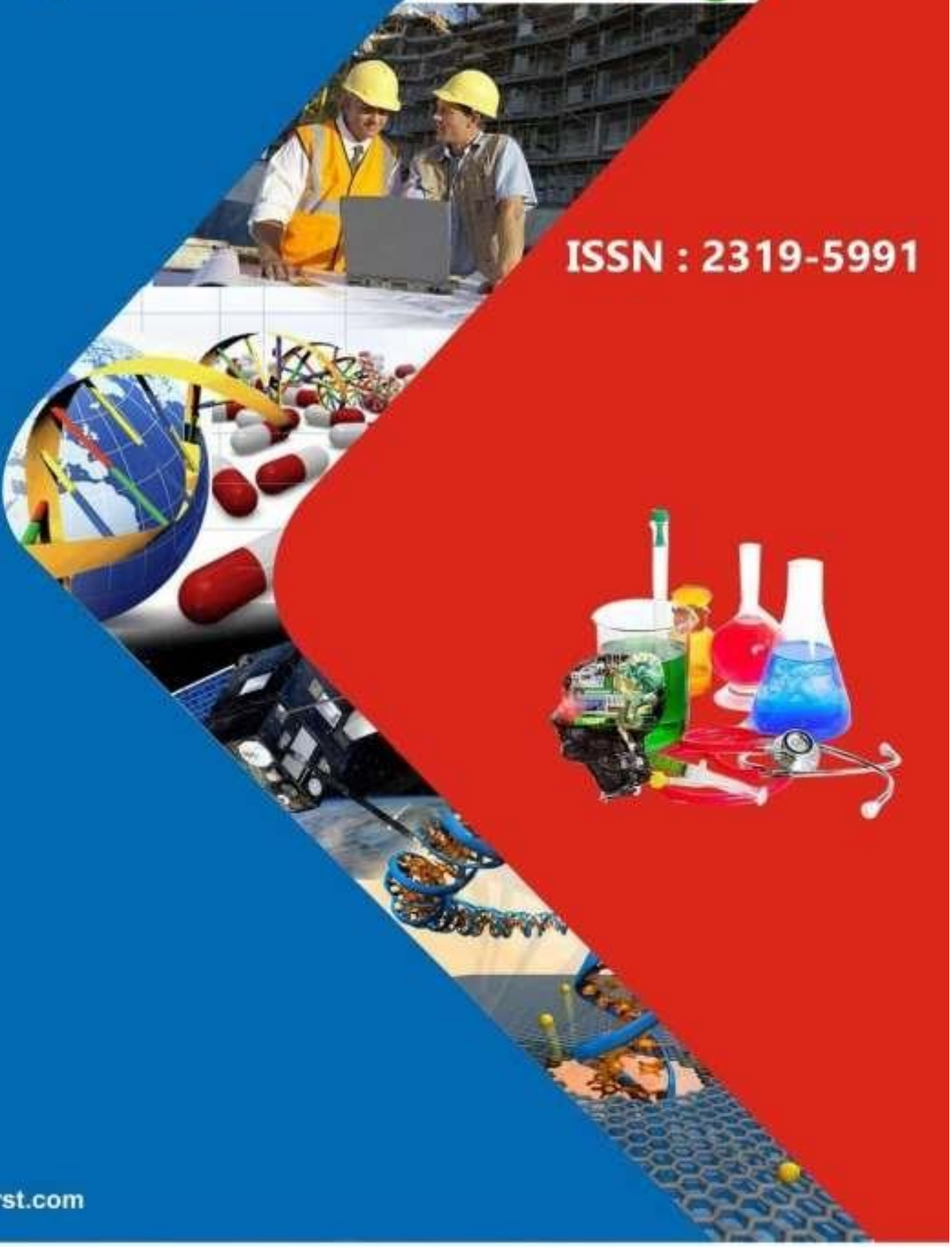
IJERST

International Journal of Engineering Research and Science & Technology

ISSN : 2319-5991



www.ijerst.com

Email: editor@ijerst.com or editor.ijerst@gmail.com

ISSN 2319-5991 www.ijerst.com

Vol. 17, Issuse.1, March 2024

SOCIAL DISTANCING USING COMPUTER VISION AND DEEP LEARING

Daram Eeshwar Prasad1, Sureddy Praveen Rao2, Sandela Shiva Sai3, V Bala 4

1,2,3 B.Tech Student, Department of CSE (Internet of Things), Malla Reddy College of Engineering and Technology, Hyderabad, India.

4 Assistant Professor , Department of CSE (Internet of Things), Malla Reddy College of Engineering and Technology, Hyderabad, India.

ABSTRACT

Social distancing is a critical measure in containing the spread of Covid-19, even with the availability of effective vaccines. To ensure the maximum reduction of virus transmission and minimize its impact, adhering to social distancing norms is imperative. In this study, we propose a Python-based deep learning approach using the YOLOv8 model trained on the COCO dataset to monitor social distancing in public spaces. Our software tool analyzes real-time video streams from CC cameras and employs OpenVINO inference for efficient and accelerated model deployment. By leveraging YOLOv8 with OpenVINO, we can accurately detect and monitor individuals' compliance with proper social distancing practices.

Keywords: Social Distancing, Deep Learning, YOLOv8, COCO, OpenVINO. Abbreviations and Acronyms: COCO-Common Objects in Context, OpenVINO-Open Visual Inference and Neural Network Optimization.

I.INTRODUCTION

When the novel coronavirus pandemic emerges, the spread of the virus has left public keep anxiety if they do not have any effective cure. The World Health Organization (WHO) has declared Covid-19 as a pandemic due to the increase in the number of cases reported around the world. To contain the pandemic, many countries have implemented a lockdown where the government enforced that the citizens to stay at home during this critical period. The public health bodies such as the Centers for Disease Control and Prevention (CDC) had to make it clear that the most effective way to slow down the spread of Covid-19 is by avoiding close contact with other people.To flatten the curve on the Covid-19 pandemic, the citizens around the world are practicing physical distancing. In reducing the impact of this coronavirus pandemic, practicing social distancing and self-isolation have been deemed as the most effective ways to break the chain of infections after restarting the economic activities. In fact, it

ISSN 2319-5991 www.ijerst.com
Vol. 17, Issuse.1, March 2024

has been observed that there are many people who are ignoring public health measures,especially with respect to social distancing.It is understandable that given the people's excitement to start working again, they sometimes tend to forget or neglect the implementation of social distancing. Hence, this work aims to facilitate the enforcement of social distancing by providing automated detection of social distance violation in workplaces and public areas using a deep learning model. In the area of machine learning and computer vision, there are different methods that can be used for object detection.

MOTIVATION BEHIND THE PROBLEM

Deliberately increasing the physical space between people to avoid spreading illness. Staying at least one meter away from other people lessens your chances of catching covid-19. Staying in the same close environment of a covid-19 patient(including workplace, classroom, household, gatherings).By leveraging advanced technologies like computer vision and deep learning, this project aims to help monitor and enforce social distancing measures in various public sectors.

PRIOR WORK

Various studies have explored object detection and person detection using deep learning techniques, but many of these works have employed unoptimized versions and relied on OpenCV. Object classification and detection using deep learning have gained significant attention, with a particular emphasis on human detection as a subset of object detection. These works typically involve localizing and classifying detected objects based on predefined models and their shapes. In the development of a social distancing analyzer tool, computer vision, deep learning, and Python were utilized to ensure safety by detecting the distance between individuals. The work employed the various models like SSDlite Mobilenet_v2, FasterRcnn, Various YOLO versions etc, which is based on convolutional neural networks, computer vision principles, and deep learning algorithms. The initial step involved detecting persons and giving the bounding boxes values, after that calculation of Euclidean distance between those boxes and giving the results as frames.

OUR APPROACH

- Model used in our project is YOLOv8 nano model. The YOLOv8 algorithm developed by Ultralytics is a cutting-edge, state-of-the-art (SOTA) model that is designed to be fast, accurate, and easy to use, making it an excellent choice for a wide range of object detection, image segmentation, and image classification tasks.
- We downloaded the YOLOv8 model using ultralytics YOLO python module and we converted the YOLOv8 pytorch model file into OpenVINO IR model using a pre-defined

ISSN 2319-5991 www.ijerst.com
Vol. 17, Issuse.1, March 2024

function such as YOLO.export() or mo.convert_model() and that converted IR model is used for inferencing.

- We further optimized the model using NNCF Post-training Quantization API, first we created a dataset and then we run the nncf.quantize() for getting an optimized model. After that we Serialize OpenVINO IR model using the openvino.runtime.seirialize().
- Now we will read the model using openvino runtime core.read_model() then we compiled that model with device compatibility as CPU.
- We feed the input data to the converted model and fetch the output data. The output data is the bounding boxes coordinates of detected persons as label set to person only.
- Using those coordinates we find the centers of the boxes and find out the Euclidean distance between those boxes. Using this distance value as threshold we find out the persons who are violating the social distance norms.
- And the coordinates which violate the threshold are stored in the list of violations, from that list of violations we take the coordinates and plot them on the output screen.
- Hence the desired output is shown on the screen and this whole process is achieved very fast therefore our input feed speed almost matches with the output feed i.e the output is almost real time, achieving which is highly appreciated in computer vision tasks.
- The output is reflected on the screen along with bounding boxes around people who violated the rules i.e threshold in this case, also with count of people who violated and this way monitoring and analyzing the risk becomes an easy task for people rather than manual monitoring which is prone to mistakes.

MODULES

Module 1: YOLOv8

YOLOv8 is the latest family of YOLO based Object Detection models from Ultralytics providing state-of-the-art performance. Leveraging the previous YOLO versions, the YOLOv8 model is faster and more accurate while providing a unified framework for training models for performing Object Detection, Instance Segmentation, and Image Classification.

Module 2: NNCF Quantization

NNCF provides a suite of advanced algorithms for Neural Networks inference optimization in OpenVINO with minimal accuracy drop. We will use 8-bit quantization in post-training mode (without the fine-tuning pipeline) to optimize YOLOv8.

Module 3: OpenVINO

OpenVINO(Open Visual Inference and Neural Network Optimization) is an open-source toolkit by Intel. It optimizes and accelerates deep learning

ISSN 2319-5991 www.ijerst.com
Vol. 17, Issuse.1, March 2024

models for inference on Intel hardware platforms. It supports various frameworks, provides optimization techniques, and offers pre-trained models. OpenVINO is accessible through multiple programming languages and facilitates efficient deployment of deep learning models.

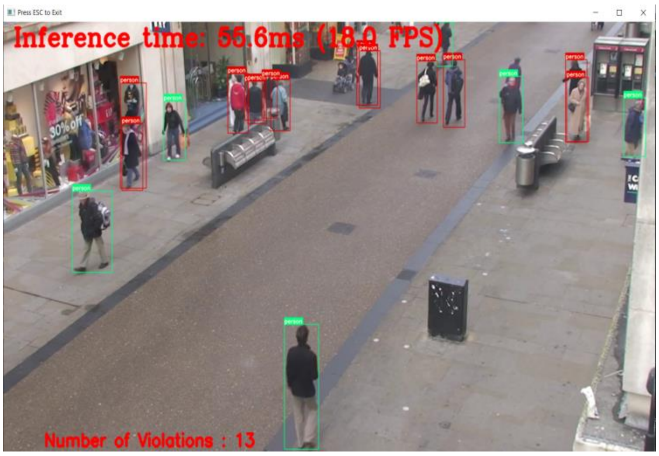
Module 4: Distance computation using Euclidean distance as metric

Computation of distance between the pairs of detected persons using Euclidean distance as metric.

$$d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

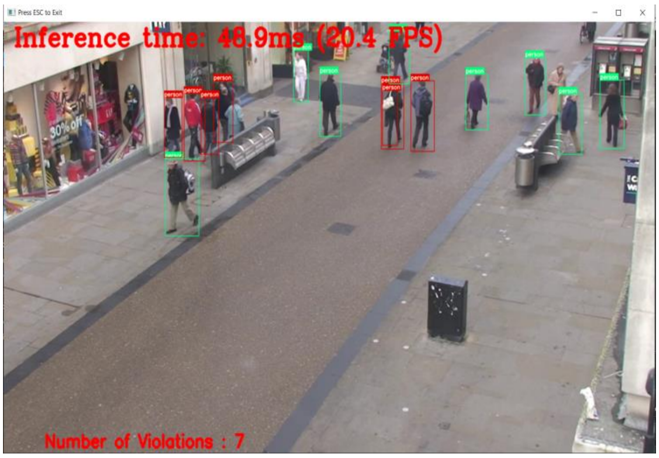
RESULTS

We provide the path of input video as an argument to the run_object_detection() function and it returns a frame of social distancing violations which will be displayed for that particular frame and violated persons will be highlighted with a good accuracy and with a good frame per seconds(FPS). The resultant video will be saved in the same directory, here the resultant video is the desired output which consists the detected frames.



Inference time: 55.6ms (18.0 FPS)
Number of Violations : 13

Fig 2: Result-1



Inference time: 48.9ms (20.4 FPS)
Number of Violations : 7

Fig 3: Result-2

CONCLUSION

Centroid tracking algorithm is used for calculating pairwise distances between the objects. To automate the process of monitoring the social distancing it is an efficient real-time deep learning based framework. The bounding boxes aid in identifying group of people satisfying the closeness property computed using pairwise vectored approach. With Euclidean distance as

ISSN 2319-5991 www.ijerst.com
Vol. 17, Issuse.1, March 2024

metric we calculated pairwise centroid distance between detected bounding boxes. The violations are displayed in the output along with violated persons.

FUTURE ENHANCEMENTS

This tool can be installed in CC cameras for monitoring social distancing in public places like malls, airports etc. We can also Use advance versions of YOLO and even more optimisation using OpenVINO for faster detections. In future and we can set the device as GPU instead of CPU which will increase speed of detection which ultimately improve the more FPS and accuracy of the model.

REFERENCES

[1]*https://docs.openvino.ai/2023.0/notebooks/230-yolov8-optimization-with-output.html*

[2]*https://docs.openvino.ai/2023.0/openvino_2_0_deployment.html*

[3]*https://docs.openvino.ai/2023.0/omz_models_group_intel.html*

[4] Z.-Q. Zhao, P. Zheng, S.-t. Xu, and X. Wu, “Object detection with deeplearning: A review,” IEEE transactions on neural networks and learningsystems, vol. 30, no. 11, pp. 3212–3232, 2019.

[5] Liquan Zhao, Shuaiyang Li, “Object Detection Algorithm Based on Improved YoloV3”, Key Laboratory of Modern Power System Simulation and Control & Renewable Energy Technology, Ministry of Education(Northeast Electric Power University), Jilin 13 2012, China.