IJNRD.ORG ISSN : 2456-4184
INTERNATIONAL JOURNAL OF NOVEL RESEARCH AND DEVELOPMENT (IJNRD) | IJNRD.ORG
An International Open Access, Peer-reviewed, Refereed Journal

Video surveillance system using yolo

Prof. Jitendra Gaikwad
Department of instrumentation and control
karma Institute of Technology Pune, India

Spandan Kamble
Department of instrumentation and control
Vishwakarma Institute of Technology
Pune, India

***Abstract*— This research paper presents an innovative application of the YOLO (you only look once) object detection framework for real-time human activity detection in video surveillance. My project focuses on four key aspects: fall detection, crash detection, social distancing detection, and object detection using CCTV cameras. Video surveillance plays an important role in every aspect of life including theft detection, unusual happening in crowded places, and monitoring suspicious activities of each individual to provide a secure and hassle-free environment.Object recognition is one of the challenging application of computer vision, which has been widely applied in many areas for e.g. autonomous cars, Robotics, Security tracking, Guiding Visually Impaired Peoples etc. With the rapid development of deep learning many algorithms were improving the relationship between video analysis and image understanding**

***Keywords*— video surveillance, fall detection, object detection, crash detection, social distancing detection.**

I. INTRODUCTION

Video surveillance has become an essential aspect of modern life, offering a sense of security and peace of mind to both individuals and businesses alike. The technology has come a long way over the years, and today it's possible to monitor and record high-quality video footage around the clock. With the help of video surveillance, we can keep an eye on our homes, offices, and public spaces, deterring criminal activity and providing valuable evidence in the event of an incident. Now a days, video surveillance has become an indispensable tool for maintaining safety and security in an ever-changing world. YOLO (you only look once) technology has ushered in a new era of smart human activity detection capabilities, has been harnessed to address a multitude of critical facets in the field of human activity monitoring. This technology has the unique ability to simultaneously detect various events and actions , including fall detection , crash detection, social distancing monitoring and object detection and the feature of YOLO lies in its capacity to perform all these predictions in a single forward pass through the neural network, rendering it only a potent tool but also remarkably fast. YOLO divides an image into a grid and several bounding boxes are formed. then a confidence score is taken for each bounding box to see whether the bounding box contains any object within it. The confidence score is high if the object inside the box matches the pre-trained YOLO dataset. The higher the confidence score, the higher the probability that a bounding box contains an object. Now several bounding boxes will intersect with each other. The more the bounding boxes intersect, more is the more the probability. The good thing about YOLO is that all the predictions in the boxes are made at the same time. In [5], Chuang et al. proposed a novel method to detect carried objects from videos using ratio histogram between two consecutive frames to identify the missing color by using more number of bins which increases computational cost. In [6], Miguel et al. fused three features based on shape and color information to detect unattended or stolen objects. In [7], Qiujie et al. developed a real-time method to detect abandoned object by using pixel-wise static region detector and color richness between the current and background frame.

II. METHODOLOGY

Under this section the important features of yolo and yolo v3 algorithms are described. [1]

A. Required software

1. Python
   language in which the code is written.
2. CMake
   For compiling openCV
3. Visual Studio
   For building openCV and darknet code
4. Nvidia GPU Driver
   for faster GPU performance
5. CUDA
   For parallel computing using GPU
6. CnDNN
   A GPU-accelerated library of primitives for deep neural networks
7. Open CV
   For working on images /videos in Python
8. Darknet
   Neural network framework for YOLO

B. Importing the libraries

After required installation is done and darknet libraries are working, we place the python files inside the [yolo\darknet\build\darknet\x64]()folder .

Data Collection: The first step in object detection and video analysis is to collect the necessary data. This can be done by capturing videos or images using cameras or accessing pre-recorded video datasets.[2]

C. working

1. working of fall detection

we take the input video from a source and divide the video into several frames. now these frames are converted into black and white. On each frame a person is detected using YOLO. now we write the code to draw rectangles on the detected persons. whenever the height of rectangle fall is not detected and when width is greater than height fall is detected. and this is how we classify the images into fall and not all fall and an alert are generated if a fall is detected. All the above process happens for a single frame. now all of this is set in a loop for each frame of the video and fall is detected.

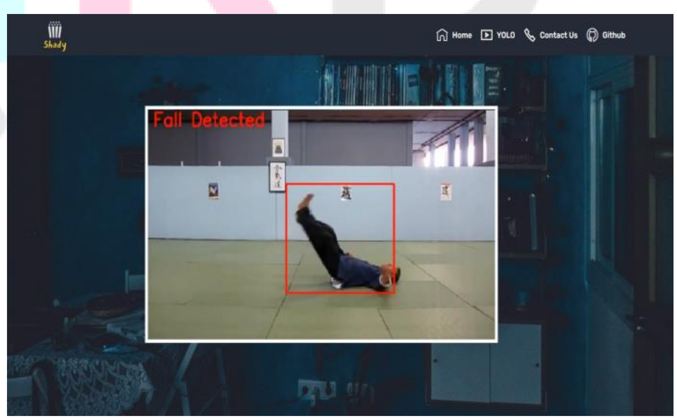

Fig 1:- Fall detected

2. social distancing detection

We take the input video from a source and divide the video into several frames. now these frames are converted into black and white. on each frame a person is detected using YOLO . now we write the code to draw rectangles on the detected persons. we check the distance between each person on the frame from each other. If the distance between the two person is less than particular value, then we colour the box red and draw a line between these boxes and add the no. of social distancing violations in a variable and display it.

all the above process happens for a single frame. now all of this is set in a loop for each frame of the video and people at risks are detected.

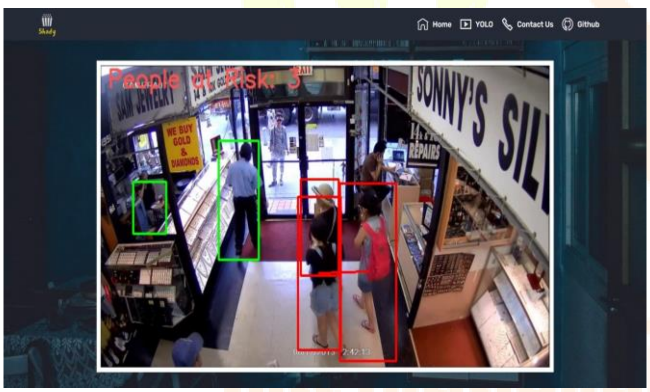

Fig 2: Social distancing detected

3. vehicle crash detection

We take the input video from a source and divide the video into several frames. Now these frames are converted into black and white. On each frame a car is detected using YOLO. now we write the code to draw rectangles on the detected cars. we check the distances between each detected car on the frame from each other. if the distance between the two cars is less than a particular value and the rectangle boxes of any two cars intersect each other then we colour the box red display the message that crash has been detected.

All the above process happens for a single frame. Now all of this is set in loop for each frame of the video and vehicle crash is detected.

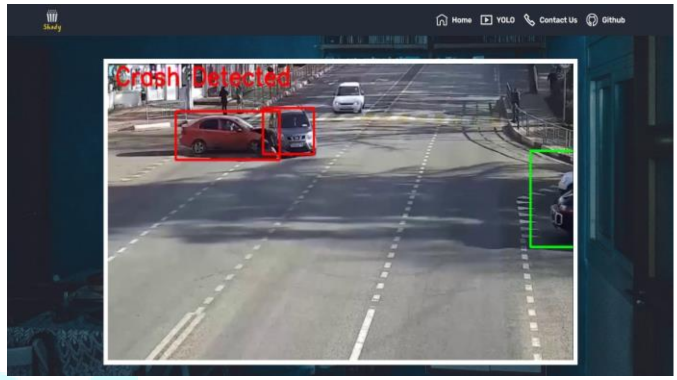

Fig 3:- crash detected

4. Object detection

We take the input video from a source and divide the video into several frames. now these frames are converted into black and white. on each frame the object is detected using YOLO.It is geerally utlizesd as a part of uses for example ,picture recovery ,security, observation , and propelled driver help frameworks.[4]

Fig 4:- Object detection

III. RESULTS

We have deployed our project using flask. when we run the script, our website is hosted onto local server, and we can use that website to perform detections on videos.

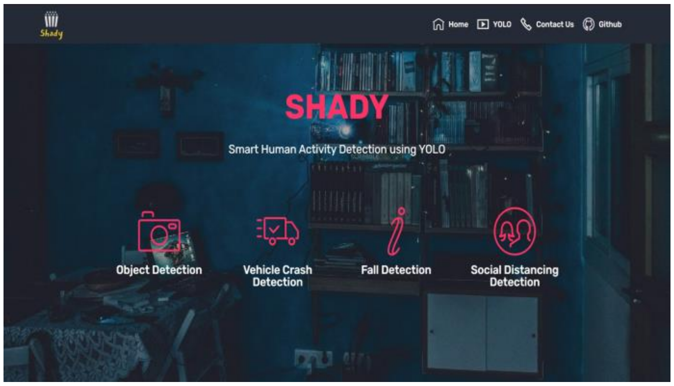

Fig 5 :- Home page

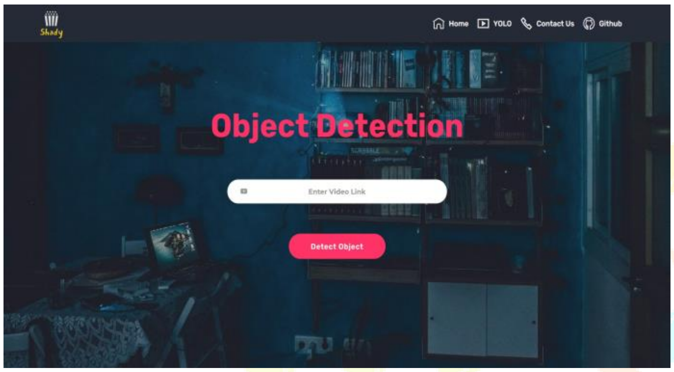

Fig 6 :- Object detection page

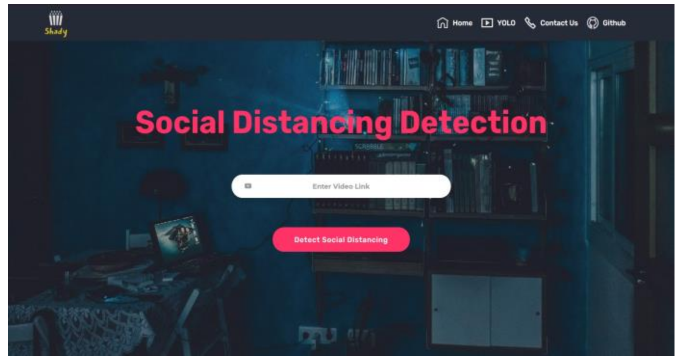

Fig 7 :- Social distancing detection page

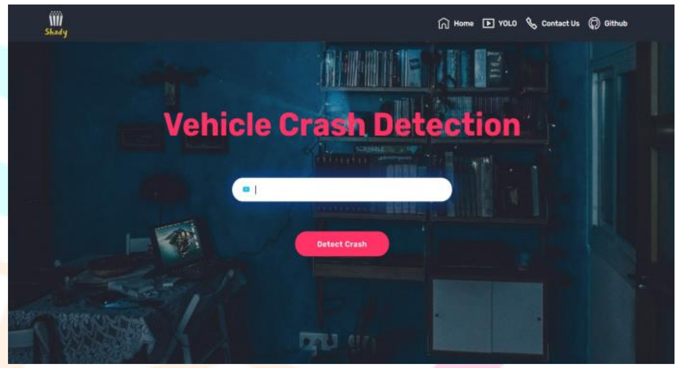

Fig 8 :- Vehicle crash detection page

IV. CONCLUSION

In this study, the paper concludes by emphasizing the significant contribution made by the research, highlighting the benefits of the integrated video surveillance system, and its potential to improve the safety in a wide range of settings. it encourages further research and development in the field of video surveillance, object detection and event recognition.

V. FUTURE SCOPE

Currently, our vehicle crash detection model is deployed locally using Flask and it uses the GPU on our computer to process the data while this setup is suitable for our current needs, it may not be scalable in the long run as we plan to deploy our website globally and reach a wider audience.

To achieve this, we plan to develop our website on cloud servers like Google Cloud and AWS, which offer GPU support. Thia swift to cloud servers will allow us to handle more traffic and provide a better user experience, as we can leverage the scalability and computational power of these cloud performs [3]. Although this would incur additional costs, it would enable us to handle more traffic and provide a better user experience.

By using cloud servers, we can also take advantage of their infrastructure, which is optimized for high performance and reliability. furthermore, deploying on the cloud would also give us the flexibility to scale up or down based on demand, ensuring that we only pay for the resources we need. In summary, while deplexperience android may involve additional costs, it would enable us to reach a wider audience, provide a better user experience, and take advantage of the cloud infrastructure scalability and reliability.

REFERENCES

[1] Mahendru, M., & Dubey, S. K. (2021). *Real Time Object Detection with Audio Feedback using Yolo vs. Yolo_v3. 2021 11th International Conference on Cloud Computing, Data Science & Engineering*

[2] Ziliani, F., & Cavallaro, A. (2001, October 1). Image Analysis for Video Surveillance Based on Spatial Regularization of a Statistical Model-Based Change Detection.

[3] Sheng, Q., Sheng, H., Gao, P., Zhu, L., & Yin, H. (2021, December 2). Real-Time Detection of Cook Assistant Overalls Based on Embedded Reasoning.

[4] Mohan ,& ravish aradhya ,H.V.(2018). simulation of object detection algorithms for video surveillance applications.

[5] C.H. Chuang, J.W. Hsieh, L.W. Tsai, S.Y. Chen and K.C. Fan, "Carried object detection using ratio Histogram and its application to suspicious event analysis," IEEE Transactions on Circuits and Systems for Video Technology, vol.19, no.6, pp. 911-916, 2009.

[6] J. C. S. Miguel and J. M. Martinez, "Robust unattended and stolen object detection by fusing simple algorithms," In proceedings of IEEE Fifth International Conference on Advanced Video and Signal Based Surveillance, pp. 18-25, 2008.

[7] Q. Li, Y. Mao, Z. Wang and W. Xiang, "Robust real-time detection of abandoned and removed objects," In proceedings of IEEE Fifth International Conference on Image and Graphics, pp. 156-161, 2009.