A Project Report On

Concentration Detection Model

Prepared by

JAY VIJAYBHAI JOSHI
Yash Subhasbhai Patel

INDEX:

Abstract:

Concentration detection models have gained significant attention as a promising solution to mitigate the risks associated with distraction-related accidents and impaired performance. These models leverage sensing technologies, data analysis algorithms, and machine learning techniques to monitor and identify signs of lack of attentiveness in individuals. This paper provides an overview of the Concentration detection model, the underlying technologies, and the challenges and opportunities it present.This project aims to contribute to enhance the safety practices and individual well-being across various domains.

Problem Specification:

Distraction while performing important tasks might poses a serious threat to public safety and individual well-being, leading to accidents, injuries, and loss of productivity. Traditional methods of Concentration detection, such as self-assessment or manual observations, are subjective and prone to errors. To address this issue, Concentration detection model have been developed. However, challenges remain in improving the accuracy and robustness of these model, managing ethical considerations regarding privacy and data usage. This report aims to explore the state-of-the-art in Concentration detection model, highlight it's capabilities, and contribute to the advancement of this field to ensure the safety and well-being of individuals in various contexts.

Introduction:

Distraction is a widespread issue that impacts people in a variety of contexts, including driving, office work, healthcare, and other places where safety is crucial. Distraction may have disastrous results, resulting in collisions, injuries, and even fatalities. Concentration detection model have become a possible remedy for this problem. In order to identify indicators of distraction in people, this model makes use of cutting-edge technology, data analysis algorithms, and machine learning approaches. Concentration detection models seek to improve safety procedures, avert accidents, and promote general wellbeing by offering real-time notifications. The purpose of this project is to shed light on the potential to transform safety procedures and protect people in diverse circumstances.

Aim & Objective:

Creation of an applied machine learning model which can detect whether a person is concentrated and attentive in critical situations like while driving, attending an important meeting or during an online examination.Moreover, creating an interface with warning which plays an warning audio whenever the person is found distracted.

Objective:

- Detecting that whether the person is Distracted or not
- Create an interface for the Concentration Detection Model
- If Distracted alert them by using the alarm sound

Connection to CS713 Assignment:

We are extending the assignment 3 by using YOLOV5 and Python in order to create our project , apart from that in this project we are using Pytorch instead of tensorflow which was used in the 2nd assingment, moreover, unlike in assignment 3, we will be able to make real time detection in this project and apart from that we will train and load the custom yolo model in our project.

Literature Review & Prior Art Search/Background:

- Benjumea, Aduen, et al. "YOLO-Z: Improving small object detection in YOLOv5 for autonomous vehicles." arXiv preprint arXiv:2112.11798 (2021).

In this paper, the authors propose YOLO-Z, a modification to the YOLOv5 object detection algorithm, specifically designed to improve small object detection in the context of autonomous vehicles. They address the challenges of detecting small objects, which are crucial for autonomous driving systems, by introducing modifications to the anchor box design, network architecture, and training strategy. The experimental results demonstrate the effectiveness of YOLO-Z in enhancing small object detection performance, making it a valuable contribution to the field of autonomous vehicles.

- Github Documentation of YOLOv5 by Ultralytics: -

https://github.com/ultralytics/yolov5 Jocher Glenn, "YOLOv5 by Ultralytics", https://github.com/ultralytics/yolov5, 2020.

This documentation on the YOLOv5 repository provides a comprehensive overview of the YOLOv5 object detection algorithm developed by Ultralytics. It covers various aspects of the algorithm, including the network architecture, training procedures, evaluation metrics, and model deployment. The documentation serves as a valuable resource for understanding the technical details and implementation of YOLOv5 in practical applications.

- Jamuna S. Murthy, G. M. Siddesh, Wen-Cheng Lai, B. D. Parameshachari, Sujata N. Patil, K. L. Hemalatha. "ObjectDetect: A Real-Time Object Detection Framework for Advanced Driver Assistant Systems Using YOLOv5." Wireless Communications and Mobile Computing, vol. 2022, Article ID 9444360, 10 pages, 2022. https://doi.org/10.1155/2022/9444360.

This research article presents ObjectDetect, a real-time object detection framework based on YOLOv5, specifically designed for Advanced Driver Assistant Systems (ADAS). The authors propose a modified network architecture and training strategy to improve the detection accuracy and efficiency for ADAS applications. The performance evaluation demonstrates the effectiveness of ObjectDetect in accurately detecting objects in real-time scenarios, making it a valuable contribution to the field of ADAS and autonomous driving.

- Mathew, M.P., Mahesh, T.Y. "Leaf-based disease detection in bell pepper plant using YOLO v5." SIViP 16, 841–847 (2022). https://doi.org/10.1007/s11760-021-02024-y.

The implementation of the YOLOv5 object identification model for leaf-based disease diagnosis in bell pepper plants is the main topic of this work. The authors provide a framework that makes use of YOLOv5 to identify and categorise several illnesses that afflict bell pepper plants based on photographs of their leaves. The usefulness of YOLOv5 in reliably detecting and diagnosing illnesses is shown by experimental findings, offering a useful tool for plant disease control in agriculture.

- Gong, Hang, et al. "Swin-Transformer-Enabled YOLOv5 with Attention Mechanism for Small Object Detection on Satellite Images." Remote Sensing 14, no. 12 (2022): 2861. https://doi.org/10.3390/rs14122861.

This research paper proposes a Swin-Transformer-Enabled YOLOv5 model with an attention mechanism for small object detection on satellite images. The authors combine the strengths of the Swin Transformer and YOLOv5 to enhance the detection performance for small objects in the challenging domain of satellite imagery.

Approach/Method:

Data Collection:

- Utilise a webcam to capture images while studying.
- Attempt to record both "distracted" and "concentrated" moments.
- Ensure that the dataset accurately depicts a range of locales, lighting situations, and physical motions.
- Gather a sufficient number of photos to expand the dataset and boost model precision.

Data Annotation:

- Utilize an annotation tool like imgLabel to annotate and label the collected images.
- Annotate each image with the corresponding class label, i.e., "distracted" or "concentrated."
- Provide detailed annotations to help the model generalize different conditions accurately.

YOLO Configuration and Training:

- Utilize YOLO v5 for object detection tasks.
- Create and load a custom model based on YOLO v5 architecture.
- Configure the YOLO model to accommodate the custom dataset and classes.
- Train the YOLO model on the custom dataset using PyTorch, optimizing the model's hyperparameters.

Real-time Integration:

- Develop a script to incorporate the trained YOLO model with real-time webcam footage.
- Process each frame from the webcam in real-time using the YOLO model.
- Make predictions for the detected objects/classes in the frame.
- Analyze the predictions to determine the level of concentration of the student.
- If a student is found to be distracted, play an audio warning to alert them.

Performance Optimization:

- Optimize the YOLO model for fast inference speed to minimize latency.
- Fine-tune the hyperparameters and experiment with different architectures to improve accuracy and speed.
- Consider using hardware acceleration techniques like GPU utilization to enhance the model's performance further.

Robustness and Model Refinement:

- Gather feedback from preliminary tests to identify shortcomings and areas for improvement.
- Continuously refine the model based on the collected feedback and evaluation results.
- Address variations in student posture and behavior by diversifying the training dataset and augmenting it with relevant images.

By following this approach, the project aims to leverage the capabilities of the YOLO object detection algorithm to discern whether a student is concentrated or distracted during a learning session. Through diligent data collection, model optimization, addressing real-time processing needs, and considering ethical considerations, a functional and efficient system can be developed to monitor and enhance student engagement in educational settings and other important contexts.

Dataset Description:

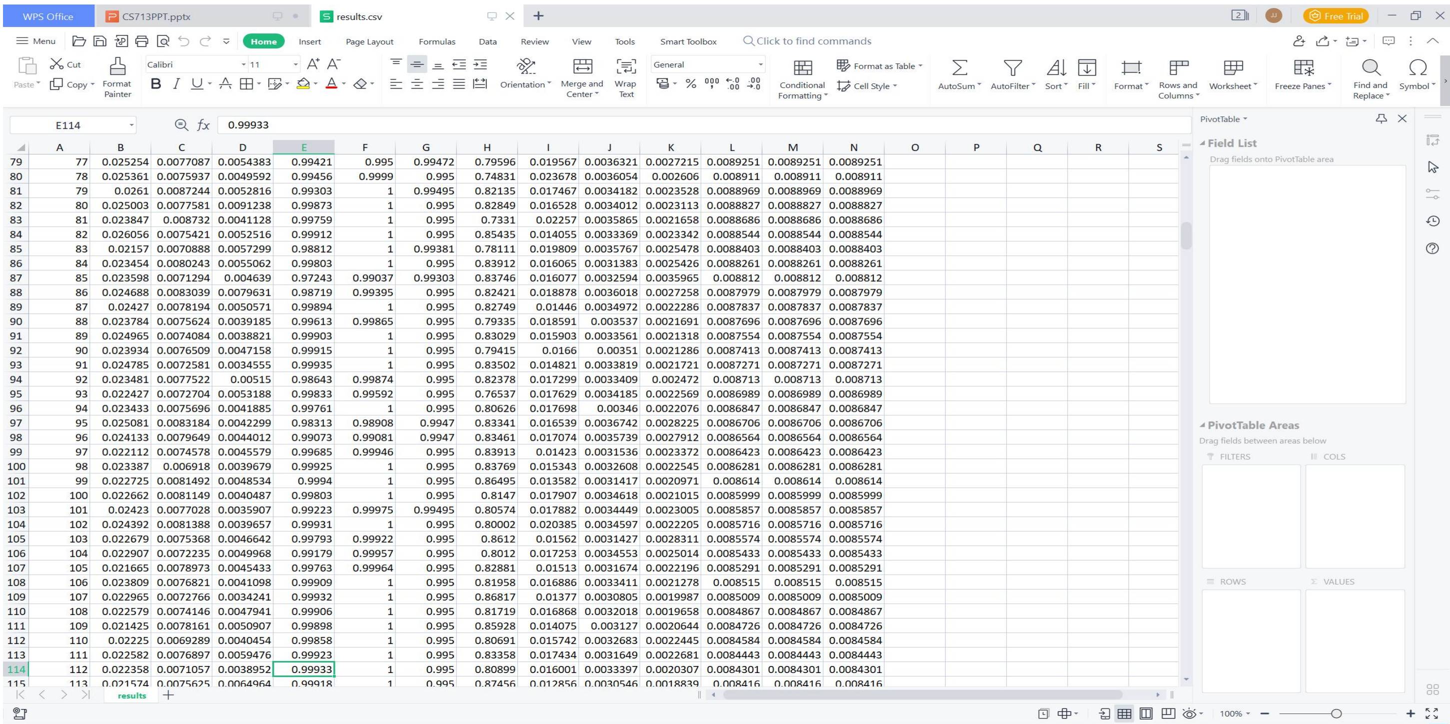
The dataset used in this study aimed to train a Concentration detection model to identify the level of concentration of students. The data collection process involved capturing image data through a webcam, resulting in a dataset comprising 209 images. The dataset was manually annotated and labeled using imgLabel, a labeling tool, to indicate two classes: 'distracted' and 'concentrated'.

One of the primary challenges encountered during the data collection phase was the unavailability of open-source datasets containing instances of the 'distracted' and 'concentrated' classes specific to this study. To overcome this limitation, images were captured directly from the webcam. This approach ensured that the dataset was tailored to the specific context and requirements of the study.

Capturing representative images that accurately depicted the states of 'distracted' and 'concentrated' was crucial for training a robust and accurate model. However, various factors, such as different lighting conditions and physical gestures, posed challenges in achieving a diverse and representative dataset. To address this issue, the data collection process was conducted in different environments and lighting conditions, aiming to capture a wide range of scenarios. Detailed annotations were provided to guide the model in understanding and generalizing the different conditions under which a student could be classified as 'distracted' or 'concentrated'.

The dataset serves as the foundation for training a YOLO model, which was configured and trained on this custom dataset. The inclusion of diverse images and accurate annotations aimed to enhance the model's ability to accurately detect and classify the level of concentration of students in real-time webcam footage. By analyzing the images captured in real time, the model could identify instances of distraction and trigger a warning system, such as playing an audio alert, to notify the individual. This monitoring system provided a means to promote and maintain concentration levels among students, ultimately contributing to improved productivity and engagement in educational settings.

Analysis:



	A	B	C	D	E	F	G	H	I	J	K	L	M	N
79	77	0.025254	0.0077087	0.0054383	0.99421	0.995	0.99472	0.79596	0.019567	0.0036321	0.0027215	0.0089251	0.0089251	0.0089251
80	78	0.025361	0.0075937	0.0049592	0.99456	0.9999	0.995	0.74831	0.023678	0.0036054	0.002606	0.008911	0.008911	0.008911
81	79	0.0261	0.0087244	0.0052816	0.99303	1	0.99495	0.82135	0.017467	0.0034182	0.0023528	0.0088969	0.0088969	0.0088969
82	80	0.025003	0.0077581	0.0091238	0.99873	1	0.995	0.82849	0.016528	0.0034012	0.0023113	0.0088827	0.0088827	0.0088827
83	81	0.023847	0.008732	0.0041128	0.99759	1	0.995	0.7331	0.02257	0.0035865	0.0021658	0.0088686	0.0088686	0.0088686
84	82	0.026056	0.0075421	0.0052516	0.99912	1	0.995	0.85435	0.014055	0.0033369	0.0023342	0.0088544	0.0088544	0.0088544
85	83	0.02157	0.0070888	0.0057299	0.98812	1	0.99381	0.78111	0.019809	0.0035767	0.0025478	0.0088403	0.0088403	0.0088403
86	84	0.023454	0.0080243	0.0055062	0.99803	1	0.995	0.83912	0.016065	0.0031383	0.0025426	0.0088261	0.0088261	0.0088261
87	85	0.023598	0.0071294	0.004639	0.97243	0.99037	0.99303	0.83746	0.016077	0.0032594	0.0035965	0.008812	0.008812	0.008812
88	86	0.024688	0.0083039	0.0079631	0.98719	0.99395	0.995	0.82421	0.018878	0.0036018	0.0027258	0.0087979	0.0087979	0.0087979
89	87	0.02427	0.0078194	0.0050571	0.99894	1	0.995	0.82749	0.01446	0.0034972	0.0022286	0.0087837	0.0087837	0.0087837
90	88	0.023784	0.0075624	0.0039185	0.99613	0.99865	0.995	0.79335	0.018591	0.003537	0.0021691	0.0087696	0.0087696	0.0087696
91	89	0.024965	0.0074084	0.0038821	0.99903	1	0.995	0.83029	0.015903	0.0033561	0.0021318	0.0087554	0.0087554	0.0087554
92	90	0.023934	0.0076509	0.0047158	0.99915	1	0.995	0.79415	0.0166	0.00351	0.0021286	0.0087413	0.0087413	0.0087413
93	91	0.024785	0.0072581	0.0034555	0.99935	1	0.995	0.83502	0.014821	0.0033819	0.0021721	0.0087271	0.0087271	0.0087271
94	92	0.023481	0.0077522	0.00515	0.98643	0.99874	0.995	0.82378	0.017299	0.0033409	0.002472	0.008713	0.008713	0.008713
95	93	0.022427	0.0072704	0.0053188	0.99833	0.99592	0.995	0.76537	0.017629	0.0034185	0.0022569	0.0086989	0.0086989	0.0086989
96	94	0.023433	0.0075696	0.0041885	0.99761	1	0.995	0.80626	0.017698	0.00346	0.0022076	0.0086847	0.0086847	0.0086847
97	95	0.025081	0.0083184	0.0042299	0.98313	0.98908	0.9947	0.83341	0.016539	0.0036742	0.0028225	0.0086706	0.0086706	0.0086706
98	96	0.024133	0.0079649	0.0044012	0.99073	0.99081	0.9947	0.83461	0.017074	0.0035739	0.0027912	0.0086564	0.0086564	0.0086564
99	97	0.022112	0.0074578	0.0045579	0.99685	0.99946	0.995	0.83913	0.01423	0.0031536	0.0023372	0.0086423	0.0086423	0.0086423
100	98	0.023387	0.006918	0.0039679	0.99925	1	0.995	0.83769	0.015343	0.0032608	0.0022545	0.0086281	0.0086281	0.0086281
101	99	0.022725	0.0081492	0.0048534	0.9994	1	0.995	0.86495	0.013582	0.0031417	0.0020971	0.008614	0.008614	0.008614
102	100	0.022662	0.0081149	0.0040487	0.99803	1	0.995	0.8147	0.017907	0.0034618	0.0021015	0.0085999	0.0085999	0.0085999
103	101	0.02423	0.0077028	0.0035907	0.99223	0.99975	0.99495	0.80574	0.017882	0.0034449	0.0023005	0.0085857	0.0085857	0.0085857
104	102	0.024392	0.0081388	0.0039657	0.99931	1	0.995	0.80002	0.020385	0.0034597	0.0022205	0.0085716	0.0085716	0.0085716
105	103	0.022679	0.0075368	0.0046642	0.99793	0.99922	0.995	0.8612	0.01562	0.0031427	0.0028311	0.0085574	0.0085574	0.0085574
106	104	0.022907	0.0072235	0.0049968	0.99179	0.99957	0.995	0.8012	0.017253	0.0034553	0.0025014	0.0085433	0.0085433	0.0085433
107	105	0.021665	0.0078973	0.0045433	0.99763	0.99964	0.995	0.82881	0.01513	0.0031674	0.0022196	0.0085291	0.0085291	0.0085291
108	106	0.023809	0.0076821	0.0041098	0.99909	1	0.995	0.81958	0.016886	0.0033411	0.0021278	0.008515	0.008515	0.008515
109	107	0.022965	0.0072766	0.0034241	0.99932	1	0.995	0.86817	0.01377	0.0030805	0.0019987	0.0085009	0.0085009	0.0085009
110	108	0.022579	0.0074146	0.0047941	0.99906	1	0.995	0.81719	0.016868	0.0032018	0.0019658	0.0084867	0.0084867	0.0084867
111	109	0.021425	0.0078161	0.0050907	0.99898	1	0.995	0.85928	0.014075	0.003127	0.0020644	0.0084726	0.0084726	0.0084726
112	110	0.02225	0.0069289	0.0040454	0.99858	1	0.995	0.80691	0.015742	0.0032683	0.0022445	0.0084584	0.0084584	0.0084584
113	111	0.022582	0.0076897	0.0059476	0.99923	1	0.995	0.83358	0.017434	0.0031649	0.0022681	0.0084443	0.0084443	0.0084443
114	112	0.022358	0.0071057	0.0038952	0.99933	1	0.995	0.80899	0.016001	0.0033397	0.0020307	0.0084301	0.0084301	0.0084301
115	113	0.021574	0.0075625	0.0064964	0.99918	1	0.995	0.87456	0.012856	0.0030546	0.0018839	0.008416	0.008416	0.008416

After completion of training we can see our results,where the columns are:-

1. epoch
2. gpu memory
3. generalized IOU ()
4. object loss
5. classification loss
6. total loss
7. targets
8. image size
9. precision
10. recall
11. mAP@.5
12. mAP@.5:.95
13. validation GIoU
14. validation object loss
15. validation class loss

The model's performance is shown by the box_loss, obj_loss, and cls_loss variables, which stand for localization accuracy, objectiveness accuracy, and class prediction accuracy, respectively.The model is learning and getting better at reliably detecting objects as training goes on, as seen by the continuous drop in all three loss values.

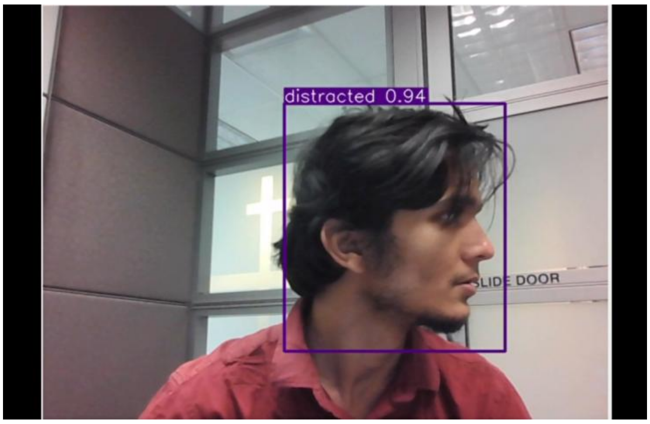
When we look at the accuracy and recall figures, we can see that both measures steadily become better with time. It may be inferred from this that the model is improving in terms of accurately detecting items and lowering false positives and false negatives.

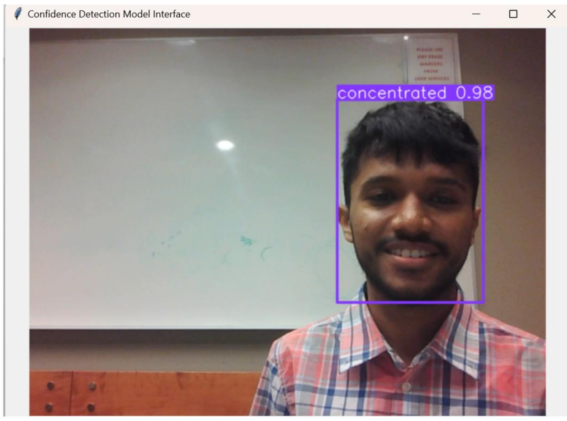
An overall assessment of the model's performance across various IoU thresholds can be made using the mAP values at different IoU thresholds (mAP_0.5 and mAP_0.5:0.95). The same as with accuracy and recall, we observe a steady increase in mAP values as the training progresses.

The values utilised to modify the weights of the model during training are represented by the learning rates (lr0, lr1, lr2).The fact that the learning rates appear to remain constant throughout the training might mean that a fixed learning rate technique is being used in the training.

Overall, the study of the data presented shows that the object detection model's performance is steadily increasing with time. The model's capacity for learning and producing accurate predictions is indicated by the declining loss values, rising accuracy, recall, and mAP values.

Results:





As we can see the Concentration Detection Model detects whether the person Concentrated or Distracted from the above two pictures also if the person is distracted th alarm will start playing, we will also add the video of the result in our git repositoryin order to showcase the capabilities of our model.

CONCLUSION:

To sum up, this project shows a creative use of object detection employing YOLO in educational and other significant situations to track and improve people's involvement. A useful and effective system was created through careful data collecting, model optimization, addressing real-time processing demands, and ethical concerns.

References:

We Referred the Following:-

- Jamuna S. Murthy, G. M. Siddesh, Wen-Cheng Lai, B. D. Parameshachari, Sujata N. Patil, K. L. Hemalatha, "ObjectDetect: A Real-Time Object Detection Framework for Advanced Driver Assistant Systems Using YOLOv5", Wireless Communications and Mobile Computing, vol. 2022, Article ID 9444360, 10 pages, 2022. https://doi.org/10.1155/2022/9444360

- Mathew, M.P., Mahesh, T.Y. Leaf-based disease detection in bell pepper plant using YOLO v5. SIViP 16, 841–847 (2022). https://doi.org/10.1007/s11760-021-02024-y.

- Gong, Hang, Tingkui Mu, Qiuxia Li, Haishan Dai, Chunlai Li, Zhiping He, Wenjing Wang, Feng Han, Abudusalamu Tuniyazi, Haoyang Li, and et al. 2022. "Swin-Transformer-Enabled YOLOv5 with Attention Mechanism for Small Object Detection on Satellite Images" Remote Sensing 14, no. 12: 2861. https://doi.org/10.3390/rs14122861

- - Benjumea, Aduen, et al. "YOLO-Z: Improving small object detection in YOLOv5 for autonomous vehicles." arXiv preprint arXiv:2112.11798 (2021).

- - Github Documentation of YOLOv5 by Ultralytics:-https://github.com/ultralytics/yolov5
  Jocher Glenn,"YOLOv5 by Ultralytics", https://github.com/ultralytics/yolov5,2020

Source Code:-

https://drive.google.com/drive/folders/1CChbS82Q_g6qAjoL4SJYEWP1O_pchOHD?usp=sharing