Model Performance Evaluation Plan – Chirag’s Section

1. Introduction

This section specifies a detailed overview of the machine learning model which will be up for evaluation.

1.1 Overview

This report provides the evaluation plan for four fall detection models. The models which are under evaluation are as the following:

1. Pose Estimation Model from Repository 1: This model uses YOLOv8 for pose estimation of elderly individuals.
2. Ambianic Fall Detection Model: This model utilizes a different approach for detecting falls, focusing on video analysis.
3. Custom YOLOv8 Model: This model is a upddated version that combines YOLOv8 with the Ambianic model to improve fall detection capabilities.
4. Garrich’s Model: This is the model which is created from scratch by the code lead of this project.

The primary aims of this evaluation is to compare the performance of these models based on several metrics, which includes their accuracy, inference time, precision, recall, F1 score, and ROC-AUC. The expected result is to identify the most effective model for fall detection.

1.2 Definitions, Acronyms, and Abbreviations

* YOLO: this stands for You Only Look Once.it is a real-time object detection system.
* Pose Estimation: this is a technique which is used to identify and trace the position of human body parts in images or in videos.
* F1 Score: this a metric that integrates precision and recall to output a single score.
* ROC-AUC: this stands for Receiver Operating Characteristic - Area Under Curve, which is a performance measurement for classification problems at numerous threshold arrangements.

3. Evaluation Metrics

The following metrics will be leveraged upon to evaluate the performance of each model:

3.1 Accuracy

Accuracy is defined as the ratio of accurately predicted instances by the model to the total instances. It is calculated through the following formula:

Accuracy = (True Positives + True Negatives)/Total Instances

3.2 Inference Time

Inference time means the time taken by the model to make the predictions based on a given input. It will be calculated in seconds for a number of test samples which would be predefined.

3.3 Precision

Precision refers to the ratio of true positive predictions to the total positive predictions. It is calculated as:

Precision= True Positives / (True Positives + False Positives) ​

3.4 Recall

Recall, which is also called sensitivity, calculates the ratio of true positive predictions to the actual positives. It is calculated as:

Recall = True Positives/ (True Positives + False Negatives)

3.5 F1 Score

The F1 score refers to the harmonic mean of precision and recall, which provides a balance between the two. It is measured as:

F1 Score = (2×Precision×Recall) / (Precision + Recall) ​

3.6 ROC-AUC

ROC-AUC is a metric that estimates the ability of the model to differentiate between classes. It is measured by plotting the true positive rate against the false positive rate at multiple thresholds and measuring the area under the curve.

6. Model Evaluation

The evaluation of the models will be directed leveraging upon the following approaches:

6.1 Evaluation on Test Data

Performance metrics will be measured on a dedicated test dataset that was not utilized at the time of training. Each of the four models will be evaluated based on the defined metrics.

6.2 Confusion Matrix

A confusion matrix will be created for each model to envision the predictions which are true positive, true negative, false positive, and false negative. This will assist in grasping the model's performance in a comprehensive manner.

6.3 ROC Curve

The ROC curve will be plotted for each model to judge the true positive rates against the false positive rates. The area under the ROC curve (AUC) will be measured to figure out the model's performance.

8. Conclusion

The evaluation plan aims to arrange a detailed examination of the four fall detection models. Based on the performance metrics, conclusions will be drawn regarding the most effective model for fall detection. On top of that, precise recommendations for further improvements or future endeavours will also be taken.

9. References

* Ultralytics (2023) Ultralytics YOLO Documentation. Available at: [https://docs.ultralytics.com](https://docs.ultralytics.com/) (Accessed: 16 August 2024).
* Ultralytics (2023) Ultralytics YOLOv8 GitHub Repository. Available at: <https://github.com/ultralytics/ultralytics> (Accessed: 16 August 2024).
* Ambianic (2023) Ambianic Fall Detection GitHub Repository. Available at: <https://github.com/ambianic/fall-detection> (Accessed: 16 August 2024).
* Gupta, A. (2021) 'Pose Estimation Using YOLO', Towards Data Science, 15 March. Available at: <https://towardsdatascience.com/pose-estimation-using-yolo-3f3c1c5a8b0d> (Accessed: 16 August 2024).
* Mubashir, M., Shao, L. and Seed, L. (2013) 'A survey on fall detection: Principles and approaches', Neurocomputing, 100, pp. 144-152. doi: 10.1016/j.neucom.2011.09.037.
* Fawcett, T. (2006) 'An introduction to ROC analysis', Pattern Recognition Letters, 27(8), pp. 861-874. doi: 10.1016/j.patrec.2005.10.010.
* Brownlee, J. (2020) 'A Gentle Introduction to Model Evaluation and Selection in Machine Learning', Machine Learning Mastery, 30 September. Available at: <https://machinelearningmastery.com/a-gentle-introduction-to-model-evaluation-and-selection-in-machine-learning/> (Accessed: 16 August 2024).