Pose Estimation with YOLOv8-pose and Kalman Filter for Tracking Falls in the Elderly

Nailfaaz
Department of Computer Science and Electronics
Gadjah Mada University
Sleman, Indonesia
nailfaaz@mail.ugm.ac.id

Abstract— **Falls are a major public health concern for the elderly, leading to serious injuries, disability, and even death. Fall detection systems can help prevent these injuries by alerting caregivers or medical professionals when a fall occurs. In this study, we introduce a novel approach utilizing the YOLOv8-pose model, an advanced human pose estimation model, combined with a Kalman Filter for tracking. The proposed system extracts both bounding boxes and pose keypoints from YOLOv8-pose. The pose keypoints are used for accurate pose estimation, while the bounding boxes are utilized for Kalman Filter-based tracking. This approach tailors a fall detection system specifically for the elderly population. By leveraging YOLOv8-pose's capabilities, we extract essential features from video data and train a classifier for fall detection and tracking. The results demonstrate that integrating YOLOv8-pose with Kalman Filter can develop effective fall detection systems for the elderly, helping to prevent injuries and improve their quality of life.**

Keywords— Fall Detection, Elderly, YOLOv8-pose, Pose Estimation, Kalman Filter, Tracking

I. Introduction

Falls among the elderly are a major public health issue, often causing serious injuries, long-term disability, and even death[1]. The World Health Organization(WHO) reports that falls are the one of the major cause of accidental or unintentional injury deaths globally[2]. For older adults, the impact of falls can be particularly severe, resulting in a loss of independence and a lower quality of life. Therefore, developing effective fall detection systems is essential to reduce these risks and ensure timely medical intervention.

Traditional fall detection methods usually involve wearable sensors or basic threshold-based algorithms. These methods can be intrusive and often lack accuracy. Recent advances in computer vision and deep learning offer promising new approaches, enabling the creation of non-intrusive, vision-based fall detection systems[5][6]. Convolutional Neural Networks (CNNs) have been widely used in this field due to their ability to accurately recognize and classify human activities from video data. However, there is increasing interest in exploring more advanced models for greater precision and reliability.

This study introduces a novel fall detection system specifically designed for the elderly, using the YOLOv8-pose model, a cutting-edge human pose estimation framework. YOLOv8-pose is excellent at detecting and tracking key points on the human body, which is crucial for identifying falls. Additionally, we integrate a Kalman Filter to improve tracking accuracy using the bounding box data from YOLOv8-pose. Combining these technologies aims to create a robust and reliable system capable of real-time fall detection.

Proposed approach involves extracting both bounding boxes and pose keypoints from the YOLOv8-pose model. The keypoints allow for precise pose estimation, while the bounding boxes enable effective tracking through the Kalman Filter. Using these features, we develop a classifier trained to accurately detect falls and track movements. The goal is to provide a non-intrusive, efficient solution that can immediately alert caregivers or medical professionals when a fall is detected, reducing the risk of serious injury and enhancing the overall quality of life for the elderly population.

II. Related Studies

Significant advancements in deep learning have greatly improved the performance of pose estimation and tracking systems. Research has shown that deep learning models, including CNNs and transformer-based architectures, significantly enhance the accuracy and robustness of these systems. These models have been successfully applied to various datasets, achieving superior results in detecting and tracking human poses even under challenging conditions such as occlusions and varying lighting. Such advancements have been instrumental in developing reliable systems for healthcare applications, particularly for monitoring the elderly.[7]

Recent studies have demonstrated the effectiveness of adapting the YOLO framework for pose estimation tasks. YOLO-Pose, for instance, has been developed to extend YOLO's capabilities for multi-person pose estimation. This method has shown promising results, achieving high accuracy on benchmark datasets and proving to be effective for real-time applications where quick and accurate human pose detection is essential.[8]

III. Literature Review

A. *YOLOv8-Pose*

YOLOv8-pose represents a significant advancement in the realm of object detection within images. Building upon the foundation laid by its predecessors, this model integrates the efficiency of the You Only Look Once (YOLO) architecture with the capability to identify human poses. YOLOv8-pose leverages the inherent strengths of CNNs, particularly in processing 2D images and extracting relevant features crucial for accurate pose estimation.[3]

The architecture of YOLOv8-pose, **Fig 1**., builds upon the foundation established by the YOLO (You Only Look Once) series while incorporating specialized components tailored for human pose estimation. At its core, YOLOv8-pose retains the efficiency and effectiveness of the YOLO architecture, characterized by its single forward pass through the neural network, enabling real-time processing of images. Its architecture consists of (1)Backbone Network: Utilizes a robust backbone network like Darknet or ResNet to extract features from input images. (2)Detection Head: Predicts bounding boxes, objectness scores, and class probabilities for detected objects. (3)Pose Estimation Module: Dedicated module for inferring human joint keypoints from detected objects, enhancing spatial context awareness. (4)Feature Fusion: Integrates features from both object detection and pose estimation stages for mutual information exchange.

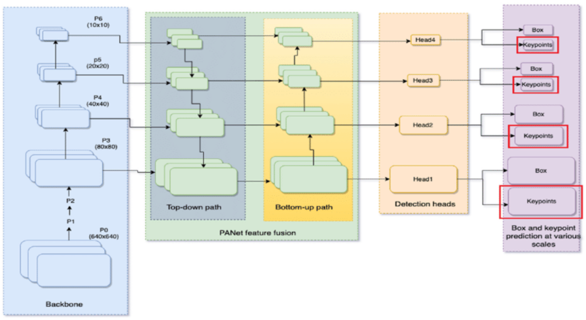

Fig. 1. YOLO-pose General Architechture.

B. *Kalman Filter*

Kalman filter is a powerful mathematical tool used for estimating the state of a dynamic system from a series of incomplete and noisy measurements. It was developed by Rudolf E. Kalman in the 1960s and has since become a fundamental technique in various fields such as navigation, robotics, and computer vision.

In terms of tracking, the Kalman filter is particularly effective because it provides a recursive solution to the problem of estimating the state of a linear dynamic system. The filter operates in two main steps: prediction and update. During the prediction step, the filter estimates the current state and its uncertainty based on the previous state and a model of the system's dynamics. In the update step, these predictions are refined using new measurements, thus correcting any discrepancies between the predicted and observed states.[4]

IV. METHODOLOGY

The methodology for the fall detection system involves a multi-step process integrating object detection, pose estimation, tracking, and fall detection algorithms.

A. *Object Detection and Pose Estimation*

Initial phase of the system leverages the YOLOv8-pose model for object detection and pose estimation. The YOLOv8-pose model is adept at detecting specific key points on the human body, which are essential for accurate pose analysis and fall detection. Key points are specific coordinates on the human body that correspond to anatomical landmarks. The YOLOv8-pose model identifies 17 key points, which are: (0)Nose, (1)Left Eye, (2)Right Eye, (3)Left Ear, (4)Right Ear, (5)Left Shoulder, (6)Right Shoulder, (7)Left Elbow, (8)Right Elbow, (9)Left Wrist, (10)Right Wrist, (11)Left Hip, (12)Right Hip, (13)Left Knee, (14)Right Knee, (15)Left Ankle, and (16)Right Ankle. Each of these key points is assigned a confidence score by the YOLOv8-pose model, indicating the likelihood of accurate detection.

In addition to detecting key points, the system establishes connections between pairs of key points to form a skeleton-like structure. These connections represent the relationships between different body parts. Each connection is visualized by drawing a line between the connected key points, provided both have confidence scores above the threshold. This visual representation helps to understand the overall pose and body orientation of the person.

Furthermore, the system calculates the bounding box that encapsulates the detected object or region of interest. By iterating over the key points and considering those above the confidence threshold, the minimum and maximum coordinates along the x and y axes are determined. This information is used to draw a rectangle around the detected person, visually enclosing the object or region of interest. This bounding box is essential for tracking and fall detection processes.

B. *Tracking Using Kalman Filter*

To maintain the identity of detected individuals across frames, I employ a Kalman Filter. When a person is detected, Kalman Filter is initialized with the center coordinates of the bounding box. Each detected person is assigned a unique track ID, and their movements are tracked across frames using the Kalman Filter. The tracker predicts the new position of each person in subsequent frames and updates the predicted position based on new detections, ensuring robust tracking even in the presence of occlusions or missing detections.

Kalman Filter also helps in associating new detections with existing tracks. By comparing the predicted positions of the tracked individuals with the positions of the new detections, the system can determine the best match for each track. This association process ensures that each person is accurately tracked over time, maintaining their unique identities and providing a reliable basis for fall detection.

C. *Fall Detection Algorithm*

The fall detection algorithm operates by assessing the speed of tracked individuals' movements, complementing the tracking mechanism facilitated by the Kalman Filter and the pose estimation capabilities of YOLOv8-pose. Rather than scrutinizing specific body orientations, this approach focuses on the velocity of key body points, particularly the head, to discern potential falls.

Upon receiving keypoints and track IDs from the pose estimation model, the Speed-Based Fall Detector calculates the speed and direction of the head movement based on consecutive positions. By comparing the current position with the previous one, the algorithm determines the speed of the head movement using Euclidean distance. Additionally, it calculates the direction of movement using the arctangent of the movement vector. If the calculated speed exceeds a predefined threshold and the direction corresponds to a fall (e.g., downwards), indicative of a sudden movement associated with a fall, the algorithm flags a potential fall event. This approach helps to prevent false positives from movements such as getting up.

The integration of this algorithm eliminates the need for extensive posture analysis and aspect ratio calculations. Instead, it prioritizes real-time assessment of movement dynamics, offering a more streamlined approach to fall detection. Additionally, by maintaining a record of previous head positions for each tracked individual, the algorithm ensures accurate speed calculations and minimizes false alarms.

V. EXPERIMENTAL RESULTS

In this experiment, we utilized a YOLO model (yolov8n-pose.pt) for pose estimation from the video file dummy.mp4. The pose estimation model identified keypoints on detected persons, which were then tracked using a Kalman filter. The Kalman filter was initialized with a state vector comprising position and velocity ([x, y, 0, 0]) and used specific matrices for state transition, observation, and noise covariances to predict and update the state of each track. The tracking employed a state covariance matrix scaled by 1000, a state transition matrix incorporating both position and velocity, an observation matrix that mapped the state to the observation space, and noise covariances for process and observation noise.

For fall detection, we implemented a Speed-Based Fall Detector with a speed threshold of 20. This detector calculated the speed of the head keypoint across consecutive frames and considered the direction of movement. Only downward movements, indicated by a direction between $\pi/4$ and $3\pi/4$ radians, were flagged as potential falls. This approach helped to prevent false positives from other movements such as getting up. The detections were associated with existing tracks using a linear assignment method with a distance threshold of 50.

Throughout the video processing, keypoints and fall detection results were logged with timestamps, track IDs, fall detection status, and speed. Sample of the detection could be seen in **Fig 2.**

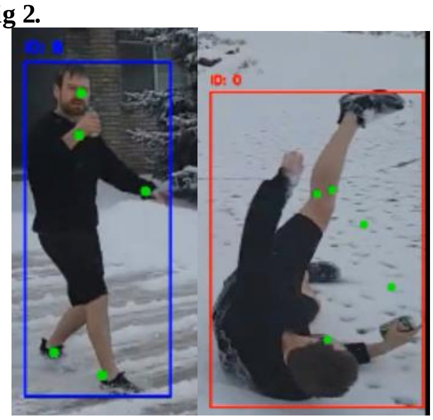

Fig. 2. Tracking, pose estimation, and fall detection on (left) no falling and (right) falling motion

In the graph depicted in **Fig 3.**, a fall is indicated by a red point where the x-coordinates are significantly higher than the y-coordinates. This pattern arises because, during a fall, an individual typically moves rapidly towards the ground, situated at the bottom of the frame. Consequently, the x-coordinate, representing horizontal position, increases, while the y-coordinate, representing vertical position, decreases sharply. It's also notable that a fall is characterized by a sudden drop in the y-coordinate. However, once the person has completely fallen and is lying on the ground, the event is no longer detected as a fall. This is because the fall detection algorithm primarily identifies the sudden change in downward speed rather than the final resting position.

In **Fig 4.**, which depicts the y-velocity over frames, the red dots primarily indicate high speeds, consistent with the rapid downward movement during a fall. However, there are also instances of high-speed points that are not red, suggesting scenarios where the individual is likely getting up rather than falling.

VI. Conclusions

Integrating the YOLOv8-pose model for precise human pose estimation with the Kalman Filter for robust tracking, developed a reliable system capable of real-time fall detection. Proposed methodology effectively distinguishes between actual falls and other rapid movements, such as getting up, by focusing on the sudden changes in downward speed and y-velocity.

The experimental results demonstrate the system's accuracy and efficiency. The YOLOv8-pose model successfully identified keypoints and bounding boxes, while the Kalman Filter maintained the identity of tracked individuals across frames. The speed-based fall detection algorithm proved effective in identifying falls without being misled by non-fall movements, reducing false positives.

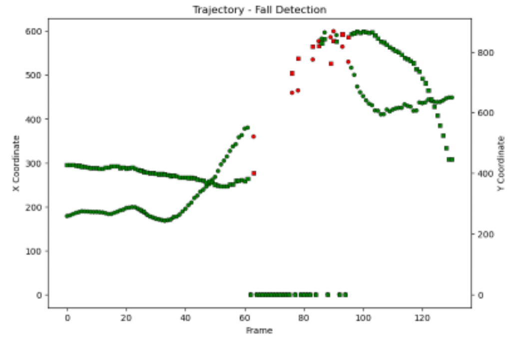

Fig. 3. Visualization of x and y coordinates over frame.

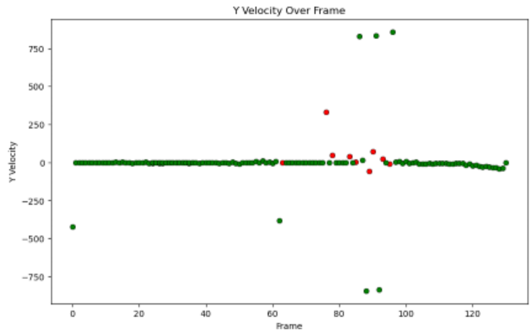

Fig. 4. Visualization of f velocity over frame.

Appendix

- Source Code: https://colab.research.google.com/drive/176k90hlLBqwSfT_hZwcYNiAIN8AhptX0?usp=sharing
(note, copy all link, don't click it)

References

[1] R. Martins, N. Carvalho, S. Batista, and A. Dinis, "Falls in elderly: Study of the prevalence and associated factors," European Journal of Development Studies, vol. 2, no. 3, pp. 12–17, May 2022. doi:10.24018/ejdevelop.2022.2.3.92

[2] "Falls," World Health Organization, https://www.who.int/news-room/fact-sheets/detail/falls (accessed May 22, 2024).

[3] Ultralytics, "Ultralytics/ultralytics: New - yolov8 in PyTorch > ONNX > OpenVINO > CoreML > TFLite," GitHub, https://github.com/ultralytics/ultralytics (accessed May 22, 2024).

[4] K. Saho, "Kalman filter for moving object tracking: Performance Analysis and filter design," Kalman Filters - Theory for Advanced Applications, Feb. 2018. doi:10.5772/intechopen.71731

[5] D. Berardini et al., "Fall detection for elderly-people monitoring using learned features and recurrent neural networks," Experimental Results, vol. 1, 2020. doi:10.1017/exp.2020.3

[6] Y. Lee, H. Yeh, K.-H. Kim, and O. Choi, "A real-time fall detection system based on the acceleration sensor of Smartphone," International Journal of Control and Automation, vol. 10, no. 1, pp. 315–326, Jan. 2017. doi:10.14257/ijca.2017.10.1.28

[7] R. B. Joshi and D. Joshi, "MoveNet: A Deep Neural Network for joint profile prediction across variable walking speeds and slopes," IEEE Transactions on Instrumentation and Measurement, vol. 70, pp. 1–11, 2021. doi:10.1109/tim.2021.3073720

[8] D. Maji, S. Nagori, M. Mathew, and D. Poddar, "Yolo-Pose: Enhancing yolo for multi person pose estimation using object keypoint similarity loss," 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Jun. 2022. doi:10.1109/cvprw56347.2022.00297