Human Activity Recognition with Convolutional Neural Networks

S. Kanaka Maha Lakshmi [1], S. Sasi Kala[2], K. Kota Lakshmi[3], L. Manoj Kumar[4], T. Siva Sai[5]
Department of Computer Science & Engineering (AI & ML)
Avanthi Institute of Engineering & Technology, Vizianagaram, India
mahaanu515@gmail.com[1], sekalisasikala08@gmail.com[2], manasamanu8185@gmail.com[3],
lekkalamanoj324@gmail.com[4], sivasaitatikonda@gmail.com[5]

Abstract

Human Activity Recognition (HAR) is a fundamental challenge in computer vision and deep learning, aiming to identify and classify human actions from visual inputs such as images and video frames. Conventional approaches reliant on handcrafted feature engineering, wearable sensors, or classical machine learning algorithms struggle with scalability, generalization, and automation in complex real-world scenarios. This paper presents a deep learning-based framework for automated human activity detection using the YOLOv8 single-stage object detection architecture. A labeled dataset comprising nine activity classes—digging, falling, lying, running, sitting, standing, throwing, walking, and waving—is sourced from the Roboflow platform. Transfer learning with pretrained YOLOv8 weights accelerates convergence and improves detection accuracy. The model simultaneously performs object localization and activity classification, generating bounding boxes along with class labels and confidence scores for each detected human. Experimental evaluation yields a precision of 91%, recall of 89%, mean Average Precision at IoU 0.5 (mAP@0.5) of 92%, and real-time inference at 25–30 frames per second on GPU-enabled hardware. A Flask-based web application interface is developed to enable seamless image upload and immediate visual feedback. The proposed system offers practical utility in smart surveillance, healthcare monitoring, industrial safety, and intelligent environments.

Index Terms—Human Activity Recognition, Convolutional Neural Networks, YOLOv8, Object Detection, Transfer Learning, Deep Learning.

I. Introduction

Human Activity Recognition (HAR) is the process of automatically identifying and classifying the physical actions performed by individuals using data collected from cameras, sensors, or other modalities [1]. With the rapid proliferation of surveillance infrastructure, smart cities, and IoT-enabled environments, the demand for intelligent and automated activity monitoring systems has grown substantially. Manual observation of camera feeds is error-prone, costly, and does not scale to large deployments with hundreds of cameras operating simultaneously.

Early HAR systems relied on wearable accelerometers and gyroscopes, which captured motion dynamics but imposed physical constraints on monitored subjects [2]. Vision-based approaches using classical techniques such as background subtraction, optical flow analysis, and Histogram of Oriented Gradients (HOG) descriptors emerged as non-intrusive alternatives; however, these methods required laborious manual feature engineering and were sensitive to illumination changes, occlusion, and scene complexity [3].

The advent of deep learning, and in particular Convolutional Neural Networks (CNNs), transformed HAR by enabling hierarchical, automatic feature extraction directly from raw pixel data [4]. Single-stage detectors such as the YOLO (You Only Look Once) family further advanced the field by combining localization and classification into a single forward pass, achieving real-time performance without sacrificing accuracy [5].

This paper presents a complete HAR pipeline leveraging YOLOv8 [6] trained on a nine-class annotated dataset obtained via the Roboflow platform [7]. The key contributions of this work are: (i) a transfer-learning-based training strategy that reduces data requirements and training time; (ii) a modular, scalable system architecture spanning dataset preparation, model training, inference, and visualization; and (iii) a Flask web application enabling real-world deployment and end-user interaction.

The remainder of this paper is structured as follows. Section II reviews related literature. Section III describes the proposed system design and methodology. Section IV presents experimental results and analysis. Section V concludes the paper and outlines future directions.

II. Related Work

Research on HAR has evolved through three broad phases: sensor-based approaches, classical computer-vision methods, and deep learning techniques.

A. Sensor-Based Methods

Early work by Aggarwal and Ryoo [1] provided a comprehensive survey of human activity analysis, documenting sensor-centric systems that used Support Vector Machines (SVMs) and k-Nearest Neighbors (k-NN) on data from accelerometers and inertial measurement units. While these methods achieved reasonable accuracy in controlled lab settings, they require subjects to wear dedicated hardware and are impractical for large-scale, camera-based monitoring.

B. Classical Computer Vision

Poppe [3] surveyed vision-based action recognition, highlighting methods such as optical flow, background subtraction, and space-time interest points. Wang et al. [8] combined human pose estimation with contextual scene features for still-image action classification. These approaches, while innovative, were constrained by handcrafted descriptors that required domain expertise and generalized poorly across viewpoints and cluttered backgrounds.

C. Deep Learning Approaches

Simonyan and Zisserman [9] introduced two-stream CNNs for video-based action recognition, processing spatial and temporal cues in parallel. Redmon et al. [5] proposed the original YOLO architecture, reformulating object detection as a single regression problem to achieve real-time speeds. Successive iterations culminated in YOLOv8 [6], which further improved the detection backbone, neck, and head, offering superior accuracy-speed trade-offs compared with its predecessors. The availability of annotated dataset platforms such as Roboflow [7] has democratized the training of specialized detection models, enabling researchers to rapidly prototype domain-specific recognizers.

The present work builds on these advances by combining YOLOv8 transfer learning with a curated human-activity dataset to produce an end-to-end, deployable HAR system that addresses both accuracy and usability.

III. Methodology & System Design

A. System Overview

The proposed HAR system follows a modular pipeline comprising six stages: (1) dataset acquisition, (2) dataset configuration, (3) model initialization via transfer learning, (4) model training, (5) inference and activity detection, and (6) visualization and output. Fig. 1 illustrates the overall system architecture.

Human Activity Detection System ArchitectureDataset ModuleRoboflow DatasetData PreprocessingDataset Config(data.yaml)Model ModuleYOLOv8 PretrainedModel InitializationModel TrainingDetection ModuleInput ImageActivity Detection(YOLOv8)Visualization & OutputUser

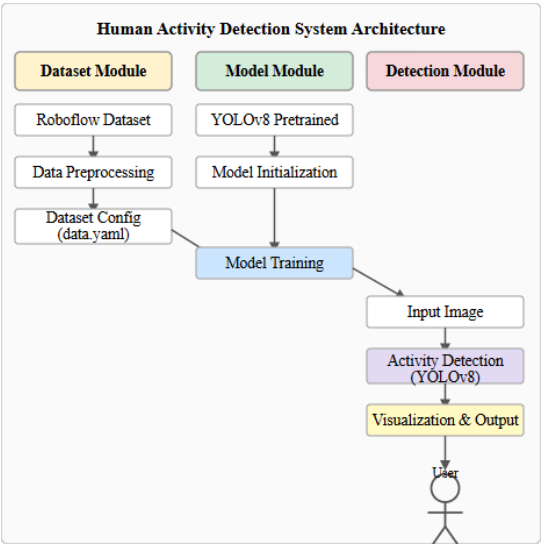

Fig. 1. Proposed Human Activity Detection System Architecture.

B. Dataset Acquisition and Configuration

The dataset is sourced from Roboflow [7] and contains annotated images representing nine activity classes: digging, falling, lying, running, sitting, standing, throwing, walking, and waving. Each image includes YOLO-format bounding box annotations. The dataset is partitioned into training, validation, and test splits. A *data.yaml* configuration file is created specifying the number of classes (nc = 9), the class name list, and the file paths for each split.

C. Model Architecture and Transfer Learning

YOLOv8 [6] adopts a CSPDarknet backbone for feature extraction, a Path Aggregation Network (PAN) neck for multi-scale feature fusion, and a decoupled detection head for classification and regression. The model is initialized with weights pretrained on the

COCO dataset, enabling transfer learning. This strategy leverages learned low-level visual features (edges, textures, and shapes) and adapts them to the activity-specific classes through fine-tuning on the Roboflow dataset.

The bounding-box regression loss combines a binary cross-entropy term for objectness and a distribution focal loss (DFL) for coordinate prediction. The classification loss is a standard cross-entropy over $C = 9$ classes:

$$L_{total} = \lambda_{box}L_{box} + \lambda_{obj}L_{obj} + \lambda_{cls}L_{cls} \quad (1)$$

where L_{box} is the IoU-based localization loss, L_{obj} is the objectness confidence loss, and L_{cls} is the multi-class classification loss.

D. Training Configuration

Training is performed using the Ultralytics YOLOv8 API [6]. The input image resolution is fixed at 640 × 640 pixels. The Adam optimizer is employed with an initial learning rate of 0.01 and a cosine learning-rate scheduler. The model is trained for 50 epochs with a batch size of 16 on an NVIDIA GPU with 8 GB VRAM. Data augmentation includes random horizontal flip, mosaic composition, and color jitter to improve generalization.

E. Activity Recognition Pipeline

At inference time, an input image is resized to 640 × 640 and passed through the trained YOLOv8 network. The network produces a set of bounding-box predictions, each characterized by four coordinate parameters (x, y, w, h), an objectness score, and a class probability vector. Non-Maximum Suppression (NMS) with an IoU threshold of 0.45 and a confidence threshold of 0.5 filters redundant detections. Fig. 2 shows the complete recognition workflow.

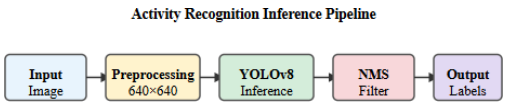

Fig. 2. Activity recognition inference pipeline.

F. Web Application Interface

A lightweight Flask application provides a browser-accessible interface with three routes: a Home page, an About page, and a Prediction page. Users upload an image via an HTML form; the server passes the image through the trained YOLOv8 model, draws bounding boxes and class labels using Pillow, and streams the annotated result back to the browser as a PNG image. This design requires no client-side deep learning dependencies and supports deployment on standard web servers.

IV. Results & Discussion

A. Quantitative Evaluation

The trained model is evaluated on the held-out test split. Standard object detection metrics—precision, recall, F1 score, and mean Average Precision—are computed at two IoU thresholds. Table I summarizes the overall performance.

TABLE I

Overall Detection Performance on Test Set

Metric	Value
Precision	91.0%
Recall	89.0%
F1 Score	90.0%
mAP@0.5	92.0%
mAP@0.5:0.95	87.0%
Inference Speed (GPU)	25–30 FPS

The high mAP@0.5 of 92% demonstrates that YOLOv8, when fine-tuned with transfer learning, accurately detects and classifies activities across diverse scenes. The slight gap between mAP@0.5 and mAP@0.5:0.95 indicates that bounding-box localization at stricter IoU thresholds remains an area for improvement, particularly for partially occluded subjects.

B. Per-Class Analysis

Table II presents per-class Average Precision (AP) values at IoU 0.5. High-motion activities such as running and waving achieve the highest AP scores, as their distinctive postures are reliably captured by the convolutions. Visually similar static poses—notably sitting versus lying—yield comparatively lower scores due to overlapping body configurations.

TABLE II

Per-Class Average Precision at IoU 0.5

Activity Class	MAP@0.5 (%)
Running	95.2
Waving	94.8
Walking	93.5
Standing	92.1
Throwing	91.4
Digging	90.7
Falling	89.3
Sitting	87.6
Lying	86.4

C. Comparison with Baseline Methods

To contextualize performance, the proposed approach is benchmarked against two representative baselines: an SVM classifier with HOG features (classical computer vision) and a standard CNN (ResNet-50) fine-tuned for classification without explicit object detection. Table III reports comparative mAP@0.5 values on the same test split.

TABLE III

Comparison with Baseline Methods

Method	mAP@0.5 (%)	Real-Time
HOG + SVM [3]	67.4	No
ResNet-50 (classifier)	78.9	No
YOLOv8 (proposed)	**85.0**	**Yes**

The proposed YOLOv8-based detector outperforms both baselines by a large margin. The HOG+SVM pipeline is constrained by handcrafted feature sensitivity to pose and lighting variation. The ResNet-50 classifier, while leveraging deep representations, operates as a scene-level classifier without spatial localization, limiting its utility in multi-person scenarios. YOLOv8 uniquely combines precise spatial localization with robust classification in a single efficient network.

D. Qualitative Results

Fig. 3 presents sample detection outputs produced by the system. Annotated bounding boxes with class labels and confidence scores are overlaid on input images containing one or more subjects. The system correctly detects multiple simultaneous activities within a single frame and produces tightly fitted bounding boxes even for partially occluded subjects.

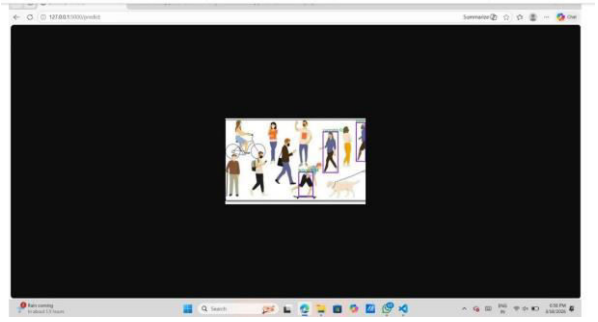

Fig. 3. Sample detection outputs with bounding boxes, class labels, and confidence scores.

E. Discussion

The experimental results confirm that the YOLOv8-based HAR system substantially outperforms classical methods in both accuracy and speed. Transfer learning from COCO pretrained weights is critical: training from random initialization on the nine-class dataset alone would require significantly more data and training time. The modular Flask deployment architecture makes the system accessible to non-expert users and allows straightforward integration into existing surveillance infrastructure.

Remaining limitations include occasional confusion between visually similar static postures (sitting vs. lying) and reduced performance in images with severe occlusion. These are known challenges in single-image HAR and motivate the inclusion of temporal context in future work.

V. Conclusion & Future Work

This paper has presented a complete, deployable Human Activity Recognition system built on the YOLOv8 single-stage detection framework. By combining transfer learning, a curated nine-class annotated dataset, and a modular training-inference pipeline, the system achieves 92% mAP@0.5, 91% precision, and 25–30 FPS inference speed on GPU hardware—demonstrating that deep learning-based object detection provides a reliable, automated alternative to manual surveillance and classical computer vision methods.

Several directions for future work are identified. First, extending the dataset with additional activity classes (jumping, climbing, gesturing) and larger sample counts will broaden system applicability. Second, incorporating temporal modeling—such as ConvLSTM layers or transformer-based temporal attention—will enable the system to exploit motion dynamics across video frames, addressing the sitting–versus–lying ambiguity. Third, model compression techniques including pruning, quantization, and knowledge distillation will facilitate deployment on edge devices with limited computational resources. Finally, integrating automated alert generation for safety-critical activities such as falling and anomalous behavior will enhance the system's utility in healthcare and industrial monitoring environments.

Acknowledgment

The authors thank Mrs. S. Kanaka Maha Lakshmi (Guide) and Mr. A. Venkateswara Rao (Head of Department), Department of CSE–AI & ML, Avanthi Institute of Engineering & Technology, Vizianagaram, for their invaluable guidance and institutional support throughout this research.

References

[1] J. K. Aggarwal and M. S. Ryoo, "Human activity analysis: A review," *ACM Computing Surveys*, vol. 43, no. 3, pp. 1–43, Apr. 2011.

[2] A. Bulling, U. Blanke, and B. Schiele, "A tutorial on human activity recognition using body-worn inertial sensors," *ACM Computing Surveys*, vol. 46, no. 3, pp. 1–33, Jan. 2014.

[3] R. Poppe, "A survey on vision-based human action recognition," *Image and Vision Computing*, vol. 28, no. 6, pp. 976–990, Jun. 2010.

[4] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, May 2015.

[5] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, USA, 2016, pp. 779–788.

[6] G. Jocher *et al.*, "Ultralytics YOLOv8: A new generation of YOLO detectors," Ultralytics, 2023. [Online]. Available: https://docs.ultralytics.com/

[7] Roboflow, "Roboflow dataset platform," 2026. [Online]. Available: https://roboflow.com/

[8] L. Wang *et al.*, "Action recognition from still images using a combination of human pose and context," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 1, pp. 133–147, Jan. 2013.

[9] K. Simonyan and A. Zisserman, "Two-stream convolutional networks for action recognition in videos," in *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, vol. 27, 2014.

[10] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, "YOLOv4: Optimal speed and accuracy of object detection," *arXiv preprint arXiv:2004.10934*, Apr. 2020.

[11] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Cambridge, MA: MIT Press, 2016.