EAI Endorsed Transactions
on AI and Robotics
Research Article EAI.EU

Advancing Public Safety with Real-Time Life Jacket Detection and Demographic Profiling Using YOLOv8 and Age Classification

Md Abu Yusuf[1,*], Nur Mohammad Chowdhury[2], Ponchanon Datta Rone[1], Partha Pratim Saha[1], Md Iqbal Hossan[1], Debabrata Sarkar[3], Ranjan Paul[1], Md Rana Hossain[1] and Madhusodan Chakraborty[1]

[1]Maharishi International University, IA, USA
[2]Louisiana Tech University, Ruston, LA, USA
[3]Chittagong University of Engineering & Technology, Bangladesh

Abstract

This study introduces a robust life jacket identification system that incorporates YOLOv8, FaceNet, and AgeNet for real-time safety surveillance in settings such as beaches, swimming pools, and maritime activities. The YOLOv8 model is applied for detecting life jackets, while FaceNet and AgeNet do face recognition and age classification, dividing persons into age groupings like "Teenager" or "Adult." The technology proficiently recognizes life jackets, detects faces, and evaluates risk by analyzing demographic factors, such as age, to generate safety alerts. The model attained a remarkable precision of 0.9934, a recall of 0.9818, and mAP50 of 0.9948, therefore validating its efficacy in recognizing life jackets and identifying individuals at risk. In high-risk aquatic situations, real-time life jacket detection, age classification, and facial recognition make the system resilient and reliable, improving public safety and risk management.

Keywords: Life Jacket, Object detection, Age Classification, Yolov8, Safety Protocol.

Received on 23 July 2025, accepted on 06 September 2025, published on 18 September 2025

doi: 10.4108/airo.9785

1. Introduction

Life jacket detection is essential to ensure safety is maintained throughout beach activities, swimming pools, and maritime operations. To avoid drowning catastrophes and provide maximum protection for safety monitoring, the real-time use of life jackets remains essential. The achievement of this goal depends heavily on object detection operations through deep learning techniques. Real-time object detection has become progressively important for various safety applications because it analyzes dynamic visual streams to detect and locate objects. This method most notably serves surveillance and autonomous system functions. YOLO (You Only Look Once) demonstrates high efficiency in real-time object detection operations due to its dual capability of fast execution and precise object detection [1] in real-time applications. YOLOv8 provides superior performance in real-time object detection operations, as it stands among the other YOLO versions. The detection capabilities of YOLOv8 for identifying objects that include life jackets become better through its architectural advancements, which incorporate advanced backbone networks with the Wise-IoU loss function. YOLOv8 provides improved detection speed alongside more precise localization due to its architectural improvements, making it an excellent match for real-time applications [2]. The system using YOLOv8 to spot life jackets delivers accurate detection of life jackets and their absence that enables prompt alerts to safeguard persons at risk. The detection system exists to stop accidents among people who could face drowning situations. It integrates FaceNet and AgeNet, which are age and face classification

*Corresponding author. Email: samba.yusuf@gmail.com

models for offering complete monitoring features. Deep Convolutional Network FaceNet functions for face detection and recognition along with its face detection functionalities. The system has the ability to identify and find faces placed in images since it is a critical feature utilized for personal identification and security monitoring purposes.

AgeNet determines the age group of detected persons by categorizing them into specified age groups. Through this set of characteristics, the system identifies life jackets and their automatic age categorization, which enhances the safety analysis process [3]. The detection of usage of life jackets and age grouping in live operations is essential to beachside safety management with pools and maritime rescue teams. The integration of YOLOv8, FaceNet, and AgeNet is a full safety surveillance system. The combined system detects life jacket usage and retrieves their age demographics from multi-source recognition data. Individual risk level comprehension depends on full context data, as some age groups need help or supervision in aquatic settings [4]. Such systems can send notices on time based on life jacket details and human age brackets to better practice safety measures. This type of system finds applications when considering public security systems since the computer's real-time enforcement of security procedures can advance people-administered enforcement of security measures. The system operates over live images and video streams to enable it to run within beaches, pools, and other bodies of water that call for real-time observation to escape mishaps [5]. The three components, YOLOv8 to recognize life jackets, FaceNet to recognize the face, and AgeNet to recognize age, constitute an end-to-end real-time security surveillance system. The system contributes to enhanced security management by virtue of its dual purpose, which involves real-time jacket recognition combined with demographic profiling of people. The combined system has better functionality in multi-functional analysis by virtue of its rapid responses and high precision, and offers secure public monitoring in dynamic environments.

2. Literature Review

Researchers examine object detection through deep learning (DL) and machine learning (ML) models for detecting life jackets together with other safety-related objects in multiple studies. Real-time object detection applications usually rely on YOLO (You Only Look Once) because of its leading ability to rapidly process image and video data. YOLO operates at real-time speeds because it performs object prediction and classification within a single forward pass, which is suitable for safety monitoring systems according to reference [6]. Real-time object detection systems have achieved multiple enhancements throughout their development up to the latest iteration of YOLOv8 that surpasses past versions. YOLOv8 implements anchor-free designs along with the combination of advanced feature fusion techniques based on Feature Pyramid Networks (FPN) and Path Aggregation Networks (PAN) within its architecture. YOLOv8 provides enhanced accuracy for detecting small objects and complex environment objects because of its new advanced architecture [7]. The detection of human faces and their age estimation using multiple models provides important contextual information that comes in handy when monitoring life jackets. The FaceNet model successfully implements a triplet loss function for Euclidean mapping of facial data, which works on multiple recognition applications [8]. AgeNet forms an essential safety monitoring system alongside these models to detect life jackets and perform age classification of people based on facial features for risk assessment purposes [4]. Researchers examine object detection through deep learning (DL) and machine learning (ML) models for detecting life jackets together with other safety-related objects in multiple studies. Real-time object detection relies heavily on YOLO (You Only Look Once) as one of its most efficient and widespread algorithms for processing images and videos. YOLO operates at real-time speeds because it performs object prediction and classification within a single forward pass, which is suitable for safety monitoring systems according to reference [9]. Real-time object detection systems have achieved multiple enhancements throughout their development up to the latest iteration of YOLOv8 that surpasses past versions. YOLOv8 implements anchor-free designs along with the combination of advanced feature fusion techniques based on Feature Pyramid Networks (FPN) and Path Aggregation Networks (PAN) within its architecture. YOLOv8 provides enhanced accuracy for detecting small objects and complex environment objects because of its new advanced architecture [10]. The detection of human faces and their age estimation using multiple models provides important contextual information that comes in handy when monitoring life jackets. The FaceNet model successfully implements a triplet loss function for Euclidean mapping of facial data, which works on multiple recognition applications [3]. AgeNet forms an essential safety monitoring system alongside these models to detect life jackets and perform age classification of people based on facial features for risk assessment purposes [11]. YOLOv8 is better and also faster in operation by utilizing three optimization techniques that take advantage of the Cosine Learning Rate Scheduler together with batch normalization and data augmentation to lower background errors [12]. Incorporating seen optimizations gives YOLOv8 the capacity to give precise outcomes in carrying out difficult detection activities, which is advantageous for safety-critical applications. The combination of YOLOv8 with FaceNet and AgeNet models provides an end-to-end solution to identify life jackets in real-time safety applications. BYOLOv8 with FaceNet and AgeNet provides businesses with a solid solution for the detection of people in peril through their robust face detection and age recognition abilities in dynamic safety monitoring systems. These models have been found effective for similar tasks, making them promising to deploy in real-time safety systems monitoring life jackets in aquatic conditions [13].

3. Methodology

The Life jacket detection system utilizes an integrated approach to detect life jackets and analyze human attributes, such as face and age, in real-time, combining data collection, preprocessing, model training, and system deployment. Initially, 451 images were captured using a smartphone and divided into training (80%, 361 images) and validation (20%, 90 images) sets. After data augmentation, which included horizontal and vertical flips, rotations, hue, saturation, brightness, and noise adjustments, the dataset was expanded to 1,083 training images and 90 validation images. Preprocessing steps, including automatic orientation correction and resizing to 640x640 pixels, ensured consistency across the data. The YOLOv8 model, chosen for its speed and efficiency in real-time object detection, was used to detect life jackets and persons. This model was fine-tuned on the augmented dataset, trained for 100 epochs, with parameters like batch size and learning rate optimized using the Adam optimizer. For face and age detection, two pre-trained models—FaceNet and AgeNet—were integrated. FaceNet, utilizing a deep CNN, detects faces by outputting bounding boxes around each face, while AgeNet predicts the age group of the person based on facial featuresAgeNet assigns the detected face to one of several predefined age buckets, such as "(0-2)", "(4-6)", "(8-12)", "(15-20)", "(25-32)", "(38-43)", "(48-53)", and "(60-100)", and classifies individuals as "Teenagers" or "Adults" based on age thresholds. Teenagers are classified as individuals in the age range of 0-20, while Adults are classified as individuals aged 25 and above. This classification is visually represented with bounding boxes around the faces, where "Teenagers" are highlighted in yellow and "Adults" in green. Simultaneously, YOLOv8 tracks persons and detects life jackets, assigning unique IDs to each person and marking those not wearing life jackets as "Risk IDs." This is crucial for safety applications, allowing the system to flag individuals who are at risk. The system processes each frame from images, videos, or live webcam feeds, where it simultaneously detects faces, estimates ages, and checks for life jackets in real-time. The system's ability to process both offline uploads (images and videos) and live webcam streams enhances its utility for environments such as beaches, pools, or other water-related areas. The system is capable of providing real-time feedback, displaying bounding boxes with labels indicating age and life jacket status. The system's integration of FaceNet and AgeNet for age classification and YOLOv8 for object detection creates a robust, efficient tool for monitoring and ensuring safety in dynamic, real-world scenarios, while hyperparameter optimization ensures high performance and accuracy.

3.1 Data preprocessing

Deep learning commences with data preprocessing as its first step of data preparation. The process of making raw data ready for analysis requires multiple steps, followed by cleaning and refining. Data quality is the top priority in developing reliable and accurate models because data quality determines model effectiveness. Preprocessing data leads to more precise and relevant outcomes by assembling data according to the needs of analysis or machine learning operations. Figure 3.3 shows step-by-step guidance for data preparation.

3.1.1 Data Cleaning

A model requires data cleaning to eliminate unwanted or inconsistent information in order to improve both accuracy and efficiency during training. The goal of this procedure is to eliminate data elimination needed to prevent patterns from being obscured and predictions from being incorrect. The data becomes more tuned for model training when redundant and distracting information is eliminated, thus allowing successful generalization to new data points. The model learns from better representative examples due to data cleaning, thus it achieves better performance alongside improved final outcomes.

3.1.2 Data Resizing

To ensure consistent input dimensions, it is essential to resize each image to a predetermined size, such as 640x640 pixels, which is suitable for both models.

3.1.3 Data Annotation

The training of machine learning object detection models requires data annotation to function effectively. Roboflow served as our tool for annotating images by drawing boundaries on life jackets worn by persons. The images received the "Wear_lifejacket" tag to signify that life jackets were present. The precise annotations covered life jacket areas exclusively to let YOLOv8 learn what constitutes a life jacket and absent life jacket cases. The precise annotation technique applies fundamental importance to developing an effective model for the precise detection of life jackets in both photos and video streams.

3.1.4 Data Augmentation

Data augmentation serves as an essential tool in training frameworks by adding manufactured versions of data that alter variations and maintain image meaning. The model gains better generalization skills because these different transformations, such as reflections and rotations alongside color modifications, offer many real-world scenarios. Data augmentation fights overfitting and improves model stability while making it better at finding life jackets in different image settings. Visual information augmentation through the training consisted of horizontal and vertical flipping, as well as 90° clockwise and counter-clockwise rotations, and random rotations from -15° to +15°. The model received horizontal and vertical shearing modifications with shear ranges from -10° to +10° to boost its capacity to recognize items from various viewing perspectives. Color enhancements, which varied hue from -17° to +17°, saturation from ±33% and brightness between -15% to +15% were applied to training images to develop environmental tolerance of the model. The dataset became more diverse through (-11% to +11%) exposure adjustments, and the application of pixel noise added up to 0.33% of pixels. The introduced augmentations protect the model from being sensitive to image features like lighting conditions and small variations or specific viewing angles, which leads to better deployment accuracy in real environments.

3.1.5 Data Distribution

The data separation process was done methodically to achieve proper training, validation, and testing conditions through independent subsets of data. After performing data augmentation, the initial 451 images from the original dataset became 1173 images compatible for training purposes and 90 images for validating the model. There were 1,083 images that represented 92% of the total images in the training dataset, whereas the remaining 8% or 90 images belonged to the validation dataset. A data split procedure helps the model receive extensive training using the majority of the data, while reserving unseen data for performance evaluation. All data was saved as part of the COCO format, which serves as the standardized format in object detection applications for both training and testing purposes.

The precise data processing procedures, comprising detailed annotation, adequate augmentation methods, and proper dataset partition, create an appropriate framework that enables the YOLOv8 model to process a diverse data collection that facilitates quick and precise life jacket detection under real-world scenarios. Multiple stable augmentation processes like rotation, together with flipping, translation, and color modification, increase model generalisation and improve its environmental performance as well as detection consistency when many applications are involved.

3.2 Model Architecture Overview

Deep learning frameworks that enable real-time object recognition are becoming increasingly popular, and YOLO is one of those frameworks. Object localization and classification tasks are combined into a single network by YOLO, which allows for simultaneous detection without the need for the traditional use of region proposal networks or sliding windows at the same time. Because of the unified framework, YOLO is able to process objects at a rapid speed, which makes it suited for applications such as robotics, autonomous driving, and surveillance. [1]

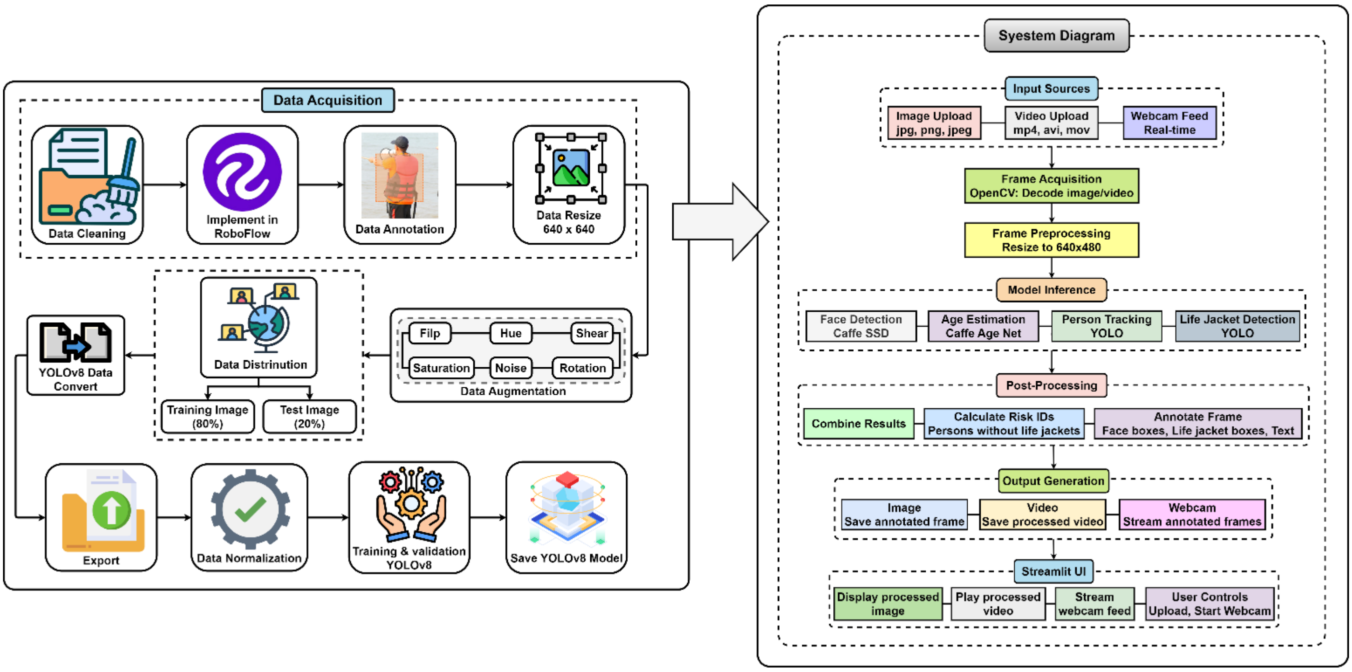

Figure 1. System Diagram Life Jacket Detection

3.2.1 YOLO Architecture Overview

Boundaries and class probabilities from image pixels are directly predicted through the Convolutional Neural Network (CNN) architecture that underlies YOLO. Each YOLO module functions in its own way for feature extraction (Backbone) and feature aggregation (Neck) as well as for executing predictions (Head). The YOLO framework treats object detection as a single-pass regression task because it needs to produce predictions for both bounding box locations and object probabilities, as well as confidence scores. [14]

Grid Division and Prediction

The YOLO model divides the input image into an $S \times S$ grid. Each grid cell is responsible for detecting objects whose center falls within that cell. For each grid cell, the model predicts multiple attributes: bounding boxes, confidence scores, and class probabilities. The image is processed as a whole, and the model makes predictions for all objects in one go. [15] For an input image of size $W \times H \times 3$ (where 3 corresponds to the RGB channels), YOLO divides it into an $S \times S$ grid. Each grid cell predicts a fixed number of bounding boxes and class probabilities. The model outputs a tensor of dimensions $S \times S \times (B \times 5 + C)$, where:

B is the number of bounding boxes each grid cell predicts (typically 2),

5 includes the 4 bounding box coordinates and 1 confidence score per box,

C is the number of object classes.

Thus, the total number of predicted values for each grid cell is $B \times 5 + C$. The confidence score (c_i) predicted by each grid cell indicates how confident the model is that the box contains an object and how accurate the predicted bounding box is. The confidence score is computed as:

$$c_i = P(\text{object}) \times \text{IoU}_{\text{pred}}^{\text{gt}}$$

Where:

$P(\text{object})$ is the probability that a given bounding box contains an object, and

$\text{IoU}_{\text{pred}}^{\text{gt}}$ is the Intersection over Union (IoU) between the predicted bounding box and the ground truth bounding box.

3.2.2 YOLOv8 Model Architecture

The architecture of YOLOv8 is an evolution of its predecessors, designed to enhance speed, accuracy, and robustness. The architecture can be divided into three main components:

Backbone

YOLOv8 extracts features from input images through its Backbone operation. The Cross-Stage Partial (CSP) architecture in YOLOv8 divides its feature map into dual segments. YOLOv8 splits its components into two parts, where the first branch applies convolution, while the second branch merges the output features of the first branch. The chosen approach elevates the learning capabilities and lowers computational requirements, thus enabling YOLOv8 to outperform its predecessors in speed and efficiency.[16]

YOLOv8 implements a C2f backbone that combines the ELAN elements from YOLOv7 with components from YOLOv5 C3 modules to enhance its structure. The combination of these architectures enables better gradient flow information collection to enhance the learning framework. [17] The backbone convolves and activates the input image to produce a collection of feature maps F_"backbone", which contain image-specific representations learned through backbone operations.

$$F_{\text{backbone}} = \text{Conv}\left(I_{\text{input}}\right)$$

Where I_{input} is the input image, and Conv denotes the convolutional operations performed on the input.

Neck

Multiple backbone layers operate in the neck layer by collecting features to enhance object identification. Multiscale feature fusion within YOLOv8 is enhanced by the implementation of Feature Pyramid Networks and Path Aggregation Networks at its neck structure. FPN in the Post-Neck and Pre-Neck architecture enables multiscale feature map generation while improving superset feature combination through spatial-rich low-level features and semantic-rich high-level features. The neck component utilizes multiple scale-level features in its operations to create F_"neck" feature maps using both scaling up and down methods [18].

$$F_{\text{neck}} = \text{FPN}(F_{\text{backbone}}) + \text{PAN}(F_{\text{backbone}})$$

This results in enhanced feature maps that are capable of detecting objects at various sizes.

Head

The Head component in YOLOv8 conducts final predictions that include bounding box outputs with confidence values, together with class prediction results. YOLOv8 divides its predictive operations into two independent branches within a decoupled head architecture that splits classification from regression tasks. Through this separate operation of tasks, the model achieves faster and more accurate predictions. YOLOv8 operates without anchor boxes, which were used in previous versions of the model. The network operates without anchors by predicting the center point before it generates bounding box coordinates directly from this position. The model becomes easier to work with because of this technique, which produces better results for small object detection. [19]

YOLOv8 concludes its analysis by generating boxes with associated class predictions along with confidence measures for all detected picture objects. The calculation to obtain the output tensor proceeds as follows:

$$\text{Output}_i = [x, y, w, h, c, P_1, P_2, \ldots, P_C]$$

Where:

x, y, w, h are the coordinates of the bounding box,

c is the confidence score for the bounding box,

$P_1, P_2, \ldots, P_C$ These are the class probabilities for each detected object.

3.2.3. YOLOv8 Loss Function

The loss function in YOLOv8 is designed to minimize errors in object localization and classification. It is composed of four main components: **Localization Loss**, **Confidence Loss**, **Classification Loss**, and **Objectless Loss**. [20]

Localization Loss

Localization loss measures how accurately the model predicts the bounding box coordinates. This is calculated using the mean squared error (MSE) between the predicted bounding box coordinates (x_i, y_i, w_i, h_i) and the ground truth coordinates $(\hat{x}_i, \hat{y}_i, \widehat{w}_i, \hat{h}_i)$:

$$\text{Localization Loss} = \sum_i \sum_j 1_{ij} \left[(x_i - \hat{x}_i)^2 + (y_i - \hat{y}_i)^2 + (w_i - \widehat{w}_i)^2 + \left(h_i - \hat{h}_i\right)^2 \right]$$

Where:

1_{ij} It is an indicator function that equals 1 if the grid cell contains an object, and 0 otherwise.

Confidence Loss

Confidence loss measures the error in predicting the confidence score c_i for each bounding box. The confidence score represents how confident the model is that the bounding box contains an object and how accurate the predicted bounding box is. This is calculated as:

$$\text{Confidence Loss} = \sum_i 1_{ij} \, (c_i - \hat{c}_i)^2$$

Where:

c_i is the predicted confidence score,

$\hat{c}_i$ is the ground truth confidence score.

Classification Loss

Classification loss measures how accurately the model predicts the class of the detected object. This is calculated using the categorical cross-entropy between the predicted class probabilities. $P_i(k)$ and the ground truth class probabilities $\hat{P}_i(k)$:

$$\text{Classification Loss} = \sum_i 1_{ij} \sum_k \left(P_i(k) - \hat{P}_i(k)\right)^2$$

Where:

$P_i(k)$ is the predicted probability of class k for grid cell i,

$\hat{P}_i(k)$ is the ground truth probability of class k.

Total Loss Function

The total loss function in YOLOv8 is the sum of the above individual losses, weighted by their respective factors:

$$\text{Total Loss} = \lambda_{\text{coord}} \cdot \text{Localization Loss} + \lambda_{\text{obj}} \cdot \text{Confidence Loss} + \lambda_{\text{noobj}} \cdot \text{Confidence Loss (no object)} + \lambda_{\text{class}} \cdot \text{Classification Loss}$$

Where:

λ_{coord}, λ_{obj}, λ_{noobj}, and λ_{class} are hyperparameters that control the relative importance of each loss component.

3.2.4 Optimizations in YOLOv8

YOLOv8 introduces several optimizations to improve detection accuracy and reduce computational load:

Wise-IoU Loss: This loss function dynamically adjusts for aspect ratio variations and scale differences, improving bounding box accuracy.

Anchor-Free Design: By eliminating anchor boxes, YOLOv8 reduces complexity and improves performance, especially for detecting small objects.

Advanced Data Augmentation: Techniques such as random cropping, color jittering, and synthetic motion blur help the model generalize better to real-world conditions.

YOLOv8 marks a major advancement within the YOLO series because it delivers advanced accuracy, fast speeds, and efficient performance. This object detection solution stands out because of its free-anchor design, together with its distinct head architecture and strong feature integration, which yield superior results in real-time scenario detection operations. The implementation of several advanced mechanisms within YOLOv8 established it as an elite object detection system that serves various applications, including autonomous vehicles and robotic surveillance systems. Real-time object detection and classification at high accuracy levels make this model an optimal selection for embedded usage as well as large-scale implementation.

3.2.5 AgeNet Model Overview

The deep learning model AgeNet serves facial image age prediction through its specific architecture. The system utilizes convolutional neural networks (CNN) to derive multiple levels of features from face pictures while determining the estimated age of the person. [21] Training takes place through regression loss functions that normally use mean squared error (MSE) to calculate the age prediction distance from the actual values. AgeNet derives its loss function through mathematical representation as:

$$\mathbf{L = \Sigma\ (y\_pred - y\_true)^2}$$

Where **y_pred** is the predicted age and **y_true** is the actual age. In our project, AgeNet is utilized to estimate the age group of individuals detected in images. By processing the facial regions identified by the FaceNet model, AgeNet classifies individuals into predefined age groups, aiding in demographic analysis and enhancing the context of life jacket detection.

3.2.6 FaceNet Model Overview

FaceNet is a facial recognition system developed by Google that learns a mapping from face images to a compact Euclidean space where distances directly correspond to a measure of face similarity. [22] The model uses a deep CNN architecture to extract facial features and employs a triplet loss function to train the network. The triplet loss function is defined as:

$$L = \Sigma\ \max(d(a\_p, a\_n) - d(a\_p, a\_a) + \alpha, 0)$$

where **a_p** is the anchor image, **a_n** is a negative image, **a_a** is a positive image, **d(·,·)** is a distance metric (e.g., Euclidean distance), and **α** is a margin that ensures a gap between positive and negative pairs. In our project, FaceNet is employed to detect and recognize faces in images. The model identifies facial features and matches them against a database to verify identity, enabling personalized alerts for individuals detected without life jackets.

3.3 Experimental Settings

A training and testing session for the model occurred through Google Colab equipment with an NVIDIA A100 GPU unit. The specifications for training the optimized YOLOv8s model, together with testing and validation, appear in Table 1. The script was written using the Python programming language, while deep learning operations used the PyTorch framework. The Roboflow API served to obtain the dataset, after which the YOLOv8s model received its training through the yolo command-line interface. The OFAT method served as a tool for hyperparameter tuning to determine the best model settings during the process. The research approach combines training models along with hyperparameter adjustment steps and performance testing, followed by outcome comparison against previous works and pre-trained models.

Table 1: Software and Hardware Details

Component	Details
OS	Windows OS
Platform	Google Colab
GPU	NVIDIA A100
Model	Yolv8s

3.4 Fine-tuning

The optimization of the YOLOv8 model included multiple hyperparameter selection procedures and adjustments to maximize its detection capabilities for life jackets. The YOLOv8s model served as the preferred choice because it maintained strength in operations and detection speed while remaining a compact form of the YOLOv8 family. The AdamW optimizer served as a training framework because it incorporates weight decay regularization with the Adam optimizer to combat both overfitting problems and improve model generalization.[23] During 100 epochs of training, there was no decay of learning rate values, which started at 0.01 and finished at 0.01. The uniform learning rate during training was established in this environment to allow dependable parameter updates of the model. The selected batch size of 16 supported memory efficiency and gradient stability throughout the training process. The images received a 640x640 pixels resize treatment to strike an equilibrium between preserving detail while minimizing computational difficulty. Automatic Mixed Precision (AMP) accelerated computation operations while reducing model performance precision, along with disabling multiscale training for maintenance of stability. A training period of 100 epochs allowed the model to adjust its weights suitably based on the specific characteristics that existed in the dataset. Model performance optimization was achieved through tuning the hyperparameters to allow the model to effective learning the required dataset features in order to develop a robust life jacket detection system. The model learned to work with different system parameters while maximizing the practical use of available computing resources during training.

Table 2: Software and Hardware Details

Hyperparameter	Value
Model	YOLOv8s
Optimizer	AdamW
Learning Rate (lr0)	0.01
Learning Rate Final (lrf)	0.01
Batch Size	16
Epochs	100
Image Size	640x640
Automatic Mixed Precision (AMP)	Enabled
Multiscale Training	Disabled
Weight Decay	0.0005
Warmup Epochs	3
Momentum	0.937
Learning Rate Scheduler	Cosine Annealing
Data Augmentation	Enabled

4. Result Analysis

The model demonstrated exceptional performance after 100 epochs of training, attaining an accuracy of 0.99334 and a recall of 0.98182, underscoring its strong capability to identify life jackets while reducing both false positives and false negatives. The mAP50 score of 0.99482 demonstrates exceptional object detection performance at a 50% Intersection over Union (IoU) threshold, whereas the mAP50-95 score of 0.85527 signifies a marginally lower yet robust detection accuracy across a wider spectrum of IoU thresholds, highlighting the model's generalization capability. The precision-recall, recall-confidence, and precision-confidence curves indicate that the model consistently exhibits excellent precision and recall across different confidence thresholds, with precision remaining near 1.0 and recall approaching 1.0 for the majority of confidence levels. The results validate the model's efficacy in precisely detecting life jackets in real-time scenarios, particularly in dynamic and safety-sensitive contexts such as beaches, swimming pools, and maritime operations. The system utilizes YOLOv8's rapid detection skills, augmented by its anchor-free architecture and sophisticated feature fusion methods (including FPN and PAN), facilitating speedy and precise object detection. The amalgamation of FaceNet and AgeNet facilitates facial recognition and age classification, offering essential context for safety monitoring by identifying individuals' age groups, thereby improving risk assessment and ensuring prompt safety alerts for vulnerable populations, such as children or the elderly. The employed process, encompassing data cleaning, augmentation, and fine-tuning of the YOLOv8 model with appropriate hyperparameters, guaranteed the model's robustness and accuracy. The amalgamation of high detection accuracy, real-time performance, and demographic profiling renders this system an optimal solution for automated safety monitoring and risk management, proficient in functioning inside real-world, high-risk scenarios.

Table 3: Model Evaluation

Metric	Score
Precision	0.9934
Recall	0.9818
MAP50	0.9948
mAP50-95	0.8552

4.1 Performance Analysis

The model demonstrated exceptional performance after 100 epochs of training, attaining an accuracy of 0.99334 and a recall of 0.98182, underscoring its strong capability to identify life jackets while reducing both false positives and false negatives. The mAP50 score of 0.99482 demonstrates exceptional object detection performance at a 50% Intersection over Union (IoU) threshold, whereas the mAP50-95 score of 0.85527 signifies a marginally lower yet robust detection accuracy across a wider spectrum of IoU thresholds, highlighting the model's generalization capability. The precision-recall, recall-confidence, and precision-confidence curves indicate that the model consistently exhibits excellent precision and recall across different confidence thresholds, with precision remaining near 1.0 and recall approaching 1.0 for the majority of confidence levels. The results validate the model's efficacy in precisely detecting life jackets in real-time scenarios, particularly in dynamic and safety-sensitive contexts such as beaches, swimming pools, and maritime operations. The system utilizes YOLOv8's rapid detection skills, augmented by its anchor-free architecture and

sophisticated feature fusion methods (including FPN and PAN), facilitating speedy and precise object detection. The amalgamation of FaceNet and AgeNet facilitates facial recognition and age classification, offering essential context for safety monitoring by identifying individuals' age groups, thereby improving risk assessment and ensuring prompt safety alerts for vulnerable populations, such as children or the elderly. The employed process, encompassing data cleaning, augmentation, and fine-tuning of the YOLOv8 model with appropriate hyperparameters, guaranteed the model's robustness and accuracy. The amalgamation of high detection accuracy, real-time performance, and demographic profiling renders this system an optimal solution for automated safety monitoring and risk management, proficient in functioning inside real-world, high-risk scenarios.

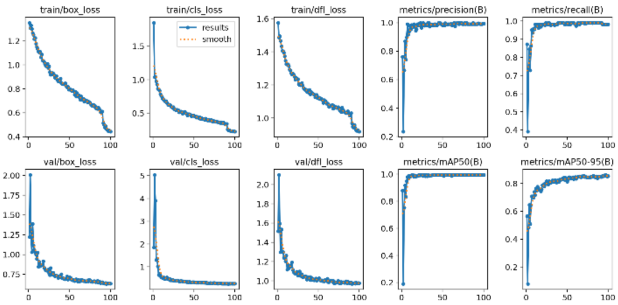

Figure 2. Performance Metric

4.1.1 Result Curve Analysis

The evaluation of the life jacket detection model is illustrated by three principal curves: Precision-Recall, Precision-Confidence, and Recall-Confidence. In the Precision-Recall Curve (a), the model demonstrates nearly flawless precision in recognizing life jackets, with a recall value nearing 1 as it recognizes the majority of true positive events. This signifies that the model is exceptionally proficient in differentiating between the "wear lifejacket" and "no lifejacket" categories. The Precision-Confidence Curve (b) demonstrates that when confidence rises, precision stabilizes at approximately 0.88 for the "wear lifejacket" class, showing the model's robust capacity to sustain precision at elevated confidence levels. Finally, the Recall-Confidence Curve (c) indicates that recall remains close to 1 for the "wear lifejacket" category, whereas it declines for the "no lifejacket" category as confidence diminishes, implying that the model excels in detecting life jackets but encounters challenges in lower-confidence situations. The curves indicate that the model demonstrates great accuracy and recall, accurately detecting life jackets in real-time with confidence criteria.

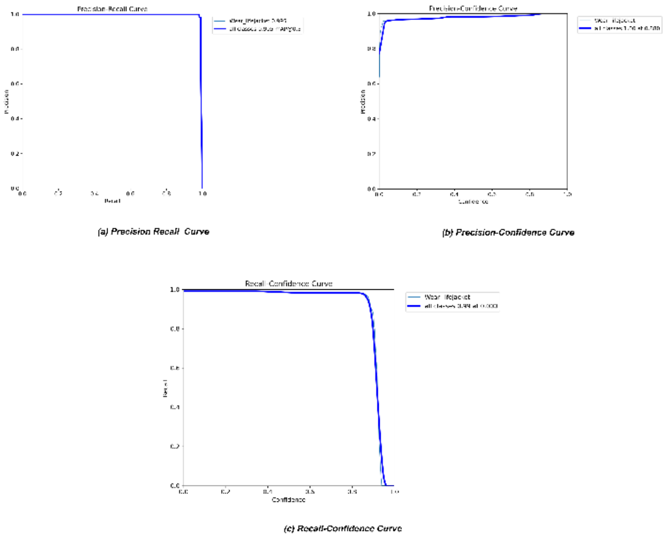

Figure 3. (a) Precision-Recall Curve, (b) Precision Confidence Curve, (c) Recall-Confidence Curve

4.1.2 Comparative analysis with other models

In this study, three object detection models—Faster R-CNN, YOLOv12, and YOLOv8—were evaluated on key performance metrics including Average Precision (AP), Precision, Recall, and mean Average Precision (mAP). The results demonstrate that Faster R-CNN achieves an AP (0.5:0.95) of 0.800, with a strong overall performance in large object detection but struggles with small objects (AP=0.000). YOLOv12 outperforms the other models, with an AP (0.5:0.95) of 0.8610, a Precision of 0.9994, and a Recall of 0.9909, showcasing superior accuracy and efficiency in object detection tasks. YOLOv8, while slightly behind YOLOv12, still demonstrates high performance with an AP (0.5:0.95) of 0.8553, a Precision of 0.9933, and a Recall of 0.9818. Furthermore, YOLOv8 offers faster training times compared to Faster R-CNN, making it more suitable for real-time applications. Overall, YOLOv12 shows the highest performance in terms of both detection accuracy and recall, followed by YOLOv8, while Faster R-CNN excels in detecting larger objects despite higher computational costs. These results provide valuable insights for selecting the appropriate object detection model based on specific use cases, balancing performance, speed, and resource requirements.

Table 4: Model Comparison

Metric / Model	Faster R-CNN	YOLOv12	YOLOv8
AP (0.5:0.95)	0.800	0.8610	0.8553

Metric / Model	Faster R-CNN	YOLOv12	YOLOv8
Precision	–	0.9994	0.9933
Recall	–	0.9909	0.9818
mAP@50 (B)	–	0.9950	0.9948
mAP@50-95 (B)	–	0.8610	0.8553
Train Loss (Final)	~1.42 (train)	Box: 0.586, Cls: 0.296, DFL: 1.043	Box: 0.442, Cls: 0.223, DFL: 0.919
Val Loss (Final)	~3.01 (val)	Box: 0.649, Cls: 0.279, DFL: 1.016	Box: 0.634, Cls: 0.257, DFL: 0.976
Training Time (Epoch 100)	~3m 15s per epoch	~5807s total	~1237s total

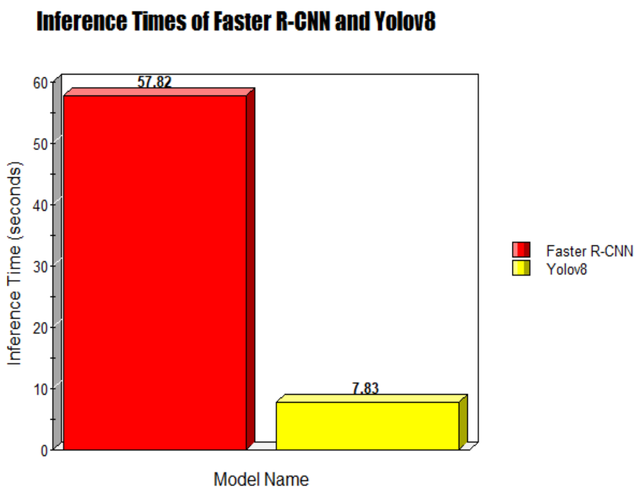

Figure 4. Comparative analysis of Inference time: Faster R-CNN Vs Yolo

4.2 Result Visualization

Figure 5. Result Inferencing

This image displays individuals wearing life jackets, each identified by a bounding box called "Wear_lifejacket." The detection system has precisely detected life jackets in diverse situations and from several angles, showcasing the model's efficacy in real-time item detection.

4.2.1 Inferencing with Age and Risk ID

Our solution incorporates the FaceNet and AgeNet models, which collaborate to identify faces and determine an individual's age. The technology not only detects if an individual is wearing a life jacket but also categorizes their age category, such as "Teenager" or "Adult." The device can identify those at risk for not wearing a life jacket and issue safety notifications.

Figure 6. Inferencing with Age classification and Risk ID in Night / Low light conditions

The integration of face detection, age classification, and life jacket status renders the system exceptionally efficient for surveillance and safety assurance, particularly in settings like beaches, pools, and boats. The amalgamation of these technologies facilitates enhanced risk evaluations and tailored safety protocols.

Figure 7. Inferencing with Age classification and Risk ID

In this figure, our system detects each person's age group, identifies whether they are wearing a life jacket, and assigns a risk ID accordingly based on their life jacket status.

4.2.2 Risk Identification and Alerts

The system first detects individuals using the FaceNet model, then classifies them as Teenagers or Adults with the AgeNet model. The Life Jacket Detection model determines if the person is wearing a life jacket, while assigning a unique ID to each individual. If a person is not wearing a life jacket, the system immediately flags them and assigns a Risk ID, triggering an alert for further action.

5. Conclusion

In summary, the YOLOv8-FaceNet-AgeNet-integrated life jacket detection system provides a dependable and highly effective way to monitor safety in real time in dynamic settings, including beaches, swimming pools, and maritime operations. The system offers complete surveillance by identifying life jackets, classifying people by age, and identifying those who are at risk by utilizing the sophisticated capabilities of YOLOv8 for quick and precise object detection, FaceNet for facial recognition, and AgeNet for age classification. The model's durability in detecting life jackets across a variety of situations is demonstrated by the outstanding performance measures, which include excellent precision, recall, and mAP scores. The system is a useful tool for improving safety procedures and guaranteeing prompt actions to prevent accidents, especially for vulnerable populations like children and the elderly, because of its real-time processing and capacity to evaluate hazards based on demographic profiling. The smooth, automated integration of face detection, age classification, and life jacket status demonstrates the system's potential for extensive use in risk management and public safety.

6. Future Scope

Future studies will focus on improving the system's efficiency and accuracy by incorporating more environmental data, like water currents or weather, to improve risk assessment. Furthermore, utilizing more sophisticated models such as multi-modal detection, which makes use of infrared or depth sensors, could enhance detection in difficult-to-reach places like busy areas or low light levels. Additionally, the system might be extended to accommodate broader uses outside of aquatic situations, like keeping an eye on public safety during outdoor events or activities. Additionally, by connecting the system's real-time monitoring features to automated alert systems or rescue services, response times and public safety could be enhanced.

7. Ethical Considerations and Privacy

All data used in this study were captured with the explicit consent of the individuals, and personal identifiers were anonymized to protect privacy. The dataset will be made publicly available upon manuscript acceptance, with strict adherence to privacy guidelines. For inferencing, publicly available images sourced from the internet were used, ensuring no personal data was violated. The system complies with GDPR and other privacy regulations, and appropriate safeguards are in place to protect individuals' rights.

References

[1] Redmon, Joseph, et al. "You only look once: Unified, real-time object detection. Proceedings of the IEEE conference on computer vision and pattern recognition. 2016.

[2] S. Ammar, T. Bouwmans, N. Zaghden, and M. Neji, "Deep detector classifier (DeepDC) for moving objects segmentation and classification in video surveillance," IET Image Process., vol. 14, no. 8, pp. 1490–1501, Jun. 2020.

[3] S. Ammar, T. Bouwmans, N. Zaghden, and N. Mahmoud, "From moving objects detection to classification and recognition: A review for smart environments," in Proc. Towards Smart World, 2020, pp. 289–316.

[4] E. M. Ibrahim, M. Mejdoub, and N. Zaghden, "Semantic analysis of moving objects in video sequences," in Proc. Int. Conf. Emerg. Technol. Intell. Syst. Bahrain: Springer, 2022, pp. 257–269.

[5] F. Ben Aissa, M. Hamdi, M. Zaied, and M. Mejdoub, "An overview of GAN-DeepFakes detection: Proposal, improvement, and evaluation," Multimedia Tools Appl., vol. 83, no. 11, pp. 32343–32365, Sep. 2023.

[6] W. Zhiqiang, L. Jun, A review of object detection based on convolutional neural network, in: 2017 36th Chinese Control Conference (CCC), 2017, pp. 11104– 11109. doi: 10.23919/ChiCC.2017.8029130.

[7] H. Ma,T.Celik, and H. Li, "Fer-YOLO: Detection and classification based on facial expressions," in Proc. Image Graphics: 11th Int. Conf. Haikou, China: Springer, 2021, pp. 28–39.

[8] D. Feng, A. Harakeh, S. L. Waslander, and K. Dietmayer, "A review and comparative study on probabilistic object detection in autonomous driving," IEEE Trans. Intell. Transp. Syst., vol. 23, no. 8, pp. 9961–9980, Aug. 2022.

[9] Arya MC, Rawat A. A review on YOLO (You Look Only One)-an algorithm for real time object detection. J Eng Sci. 2020;11:554-7.

[10] K. Tong, Y. Wu, and F. Zhou, "Recent advances in small object detection based on deep learning: A review," ImageVis.Comput.,vol.97, May 2020, Art. no. 103910.

[11] M. Hussain, H. Al-Aqrabi, M. Munawar, R. Hill, and S. Parkinson, "Exudate regeneration for automated exudate detection in retinal fundus images," IEEE Access, vol. 11, pp. 83934–83945, 2022.

[12] S. A. Singh and K. A. Desai, "Automated surface defect detection framework using machine vision and convolutional neural networks," J. Intell. Manuf., vol. 34, no. 4, pp. 1995–2011, Apr. 2023.

[13] D. Weichert, P. Link, A. Stoll, S. Rüping, S. Ihlenfeldt, and S. Wrobel, "A review of machine learning for the optimization of production processes," Int. J. Adv. Manuf. Technol., vol. 104, nos. 5–8, pp. 1889–1902, Oct. 2019.

[14] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," arXiv, 2016. [Online]. Available: https://arxiv.org/abs/1506.02640.

[15] F. Rehman, M. Rehman, M. Anjum, and A. Hussain, "Optimized YOLOV8: An efficient underwater litter detection using deep learning," Ain Shams Engineering Journal, vol. 16, 103227, 2025. [Online]. Available: https://doi.org/10.1016/j.asej.2024.103227.

[16] M. Hussain, "YOLOv5, YOLOv8 and YOLOv10: The Go-To Detectors for Real-time Vision," arXiv preprint arXiv:2407.02988v1, Jul. 2024. [Online]. Available: https://arxiv.org/pdf/2407.02988v1.

[17] X. Chen, Z. Jiao, and Y. Liu, "Improved YOLOv8n based helmet wearing inspection method," Scientific Reports, vol. 15, no. 1945, 2025. [Online]. Available: https://doi.org/10.1038/s41598-024-84555-1.

[18] M. Abu Yusuf, M. Rezaul Karim Khan, P. Pratim Saha and M. Mahbubur Rahaman, "Data Fusion of Semantic and Depth Information in the Context of Object Detection," 2024 Second International Conference on Intelligent Cyber Physical Systems and Internet of Things (ICoICI), Coimbatore, India, 2024, pp. 1124-1129, doi: 10.1109/ICoICI62503.2024.10696627.

[19] Jearanai S, Wangkulangkul P, Sae-Lim W, Cheewatanakornkul S. Development of a deep learning model for safe direct optical trocar insertion in minimally invasive surgery: an innovative method to prevent trocar injuries. Surg Endosc 2023;37(9):7295–304. https://doi.org/10.1007/s00464-023-10309-1.

[20] J. Terven, D.-M. Córdova-Esparza, and J.-A. Romero-González, "A Comprehensive Review of YOLO Architectures in Computer Vision: From YOLOv1 to YOLOv8 and YOLO-NAS," Machine Learning and Knowledge Extraction, vol. 5, no. 4, pp. 1680–1716, Nov. 2023. [Online]. Available: https://doi.org/10.3390/make5040083.

[21] D. Xu, W. Xu, and Y. Zhang, "Age Estimation Using Deep Learning and a Large-scale Dataset," Proceedings of the IEEE International Conference on Computer Vision (ICCV), 2017, pp. 1–9. [Online]. Available: https://arxiv.org/abs/1702.03810

[22] F. Schroff, D. Kalenichenko, and J. Philbin, "FaceNet: A Unified Embedding for Face Recognition and Clustering," Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015, pp. 815–823. [Online]. Available: https://arxiv.org/abs/1503.03832

[23] Ju RY, Cai W. Fracture detection in pediatric wrist trauma X-ray images using YOLOv8 algorithm. Sci Rep 2023;13(1):1–13. https://doi.org/10.1038/s41598-023-47460-7.