TABLE OF CONTENTS

CHAPTER - 1

INTRODUCTION

1.1 PROJECT OVERVIEW

Swimming is an aquatic sport. But in the water beginners feel hard to breathe which results in drowning accident. Swimmers who have mastered swimming skills will also drown when they suffer from sudden cramps and stress. This paper proposes YOLO algorithm for swimmer behavior recognition on the basis of underwater camera. This method identifies the swimmer's behavior and drowning behavior through real-time acquisition of underwater related images and alarm and data of the drowning behavior. Here, underwater camera monitoring the swimmer posture. Whereas, drowning also occurs when the victim had no intention of going into the water. However, in this project it would be able to detect near drowning using at least 1 but not more than 5 seconds of video sequence with no false positives.

1.2 PURPOSE

This system is to be implemented along the swimming pools to save human life. By studying body movement patterns and connecting cameras to artificial intelligence (AI) systems we can devise an underwater pool safety system that reduces the risk of drowning. Usually, such systems can be developed by installing more than 16 cameras underwater and ceiling and analyzing the video feeds to detect any anomalies. but AS a POC we make use of one camera that streams the video underwater and analyses the position of swimmers to assess the probability of drowning, if it is higher then an alert will be generated to attract lifeguards' attention.

CHAPTER - 2
LITERATURE SURVEY

2.1 EXISTING PROBLEM

Underwater cameras have the drawback of missing the early struggle above the water. Early on, failure to recognize a drowning scene could result in a longer rescue time, which is a significant issue to consider in a time-critical emergency. The main disadvantage of a wearable-based system is the discomfort of use. Vision-based systems and wearable sensor-based systems are two types of existing drowning detection technologies.Vision-based technologies are further subdivided into those that use underwater cameras and those that use above-water cameras.The detection accuracy of drowning person is not accurate and fast.

2.2 SURVEY WORK

2.2.1 Drowning behaviour detection in swimming pool based on deep learning (fei lei et al,2022)

In order to quickly help lifesavers judge whether people are drowning in the swimming pool, this paper proposes one efficient behavior recognition approach by means of video sequences of underwater. First, by analyzing the spatial distribution of swimming pool when swimmers are normally swimming, the data labelling and swimmer detection methods are determined. Second, a behavior recognition framework of swimmers on the basis of YOLOv4 algorithm (BR-YOLOv4) is proposed in this paper. The spatial relationship between the location information of the target and swimming/drowning area of swimming pool is analyzed to further determine the swimmer’s drowning or

swimming behavior. All the results show that the method proposed in this paper meets the Realtime detection requirements and does well in swimmer behavior recognition and provides technical support for reducing drowning accidents in public swimming pools.

2.2.2 Detection of early dangerous state in deep water of indoor swimming pool based on surveillance video (fan wang et al,2022)

Early detection of dangerous condition in the deep-water zone of swimming pool based on surveillance. This paper proposes feature extraction, feature expression and assessment criteria, including a method for evaluating normal swimming speed based on the time series of swimmers, a method for assessing an upright state that is not limited by the camera angle, and the rules for assessing state. They collected real-life data from the swimming pool and conducted related experiments. This method can easily and efficiently detect the swimmer who is in danger at an early stage and provide necessary rescue reminders to lifeguards.

2.2.3 Development of internet of things (iot) based anti-drowning device (etus et al ,2022)

This study targeted on the event of associate degree IoT based anti-drowning device to scale back the loss of lives to drowning. Agile methodology was adopted for this work. the planning and its implementation created a wristband transmitter strap and attendant alert modules with a pulse reader, a GPS huntsman, Arduino Nano and professional mini, and a red liquid substance to find a drowning person exactly. The device schematic was simulated on

Proteus software system and coded mistreatment Arduino IDE. The elements were coupled and tested, and therefore the results showed that abnormal heartbeats between 0-60 and higher than one hundred twenty triggered associate degree alert for help. The system desires a stable net affiliation for its operations and is deployed to immediate watching,time period chase, and fast location of victims.

2.2.4 Computer vision enabled drowning detection system (handalage et al, 2021)

The current systems expected to handle the matter of guaranteeing safety at swimming pools have vital issues thanks to their technical aspects, like underwater cameras and method aspects like the requirement for human intervention within the rescue mission. the utilization of an automatic visual-based observation system will facilitate to scale back drownings and assure pool safety effectively. This study introduces a revolutionary technology that identifies drowning victims in an exceedingly minimum quantity of your time and dispatches an automatic drone to save lots of them. victimization convolutional neural network (CNN) models, it will discover a drowning person Whenever such a scenario like this is often detected, the expansive tube-mounted self-driven drone can proceed a rescue mission, sounding AN alarm to tell the close lifeguards. The system conjointly keeps an eye fixed out for doubtless dangerous actions that would lead to drowning. This system's ability to save lots of a drowning victim in below a second has been incontestable in epitome experiments' performance evaluations.

2.2.5 Automated and intelligent system for monitoring swimming pool safety based on the iot and transfer learning (alotaibi, aziz et al, 2020)

Integrating the net of Things and laptop vision has been utilized in pool machine-controlled police work systems. many studies are projected to beat off-time police work drowning incidents supported employing a sequence of videos to trace human motion and position. This paper proposes Associate in Nursing economical and reliable discovering system that utilizes one image to detect and classify drowning objects, to stop drowning incidents. The projected system utilizes the IoT Associate in Nursing transfer learning to produce an intelligent and automatic answer for off-time watching pool safety. additionally, a specialized transfer learning-based model utilizing a model pretrained on "ImageNet", which may extract the foremost helpful and sophisticated options of the captured image to differentiate between humans, animals, and alternative objects, has been projected. the most aims of this is often to scale back human intervention by process and causing the classification results to the owner's mobile device

2.2.6 Automated vision-based surveillance system to detect drowning incidents in swimming pools (alshbatat et al, 2020)

This paper projected a period system which will track swimmers during a pool victimization machine learning techniques and prevents drowning accidents is projected. The system consists of a Raspberry Pi with the Raspbian software, a Pixy camera, associate degree Arduino Nano board, stepper motors, associate degree device, and motor drivers. The projected system relies on the

colourbased algorithmic rule to position and rescue swimmers United Nations agency are drowning. The device then sends associate degree alarm to the lifeguards. The results from experiments indicate that the system incorporates a distinctive capability to watch and track swimmers, thereby sanctionative it to mitigate and curb the quantity of deaths by drowning.

2.2.7 The swimmers motion detection using improved vibe algorithm (hayat et al,2019)

Swimming is one of the best exercises which helps to reduce stress. However, Swimmers may difficult to breath because of lose balance or face difficulty because of lack of training and so on leads to drowning and often leads to death. So, many researchers tried inventions to detect the drowning person but their accuracy is not up to the mark. This paper proposed a swimmer’s motion detection based on motion detection algorithm (VIBE algorithm) to detect the drowning person. An improved VIBE swimmer detection algorithm is proposed, and the algorithm is used to determine the swimmer's position. First, Images captured by a camera then change the images into Gary scale images after that background model initializing then background judgment technique and update background Model and foreground feature determination. When the moving target exists in the first frame, the improved VIBE base target detection algorithm eliminates the ghosting noise and processed to detect the person. This algorithm detects the drowning person with exact position but it still needed to improve some deficiencies.

2.2.8 A novel drowning detection method for safety of swimmers (roy et al 2018)

Effective drowning detection strategies square measure essential for the security of swimmers. during this paper, a completely unique sort of drowning detection methodology addressing several limitations of prevailing drowning detectors is projected. The projected methodology ensures detection of drowning and coverage at the sooner stages. The projected drowning detection methodology is additionally a generic answer that suites totally different water bodies from pools to oceans, associate degreed an economically viable methodology helpful for each low- and middle-income countries. The example of the drowning detection methodology is developed and incontestable and model of the system is simulated in Proteus style suite. The results of the simulation and hardware experimentation are rumored.

2.3 PROBLEM STATEMENT DEFINITION

Problem Statement (PS)	I am (Customer)	I'm trying to	But	Because	Which makes me feel
PS-1	Swimmer (Beginner)	Learn swimming	I am afraid of drowning	Lack of Training	Fear
PS-2	Lifeguard	Monitor & save swimmers	It is a difficult task to monitor	I can't able to monitor all the people at the same time	Burden & stress
PS-3	Swimmer	Practice swimming	I am afraid of drowning	Loss of balance or consciousness sometimes	Panic
PS-4	Trainer	Teach swimming	I can't able to pay attention to all learners	I can't monitor all the learners at the same time	Humiliated

CHAPTER - 3

IDEATION & PROPOSED SOLUTION

3.1 EMPATHY MAP CANVAS

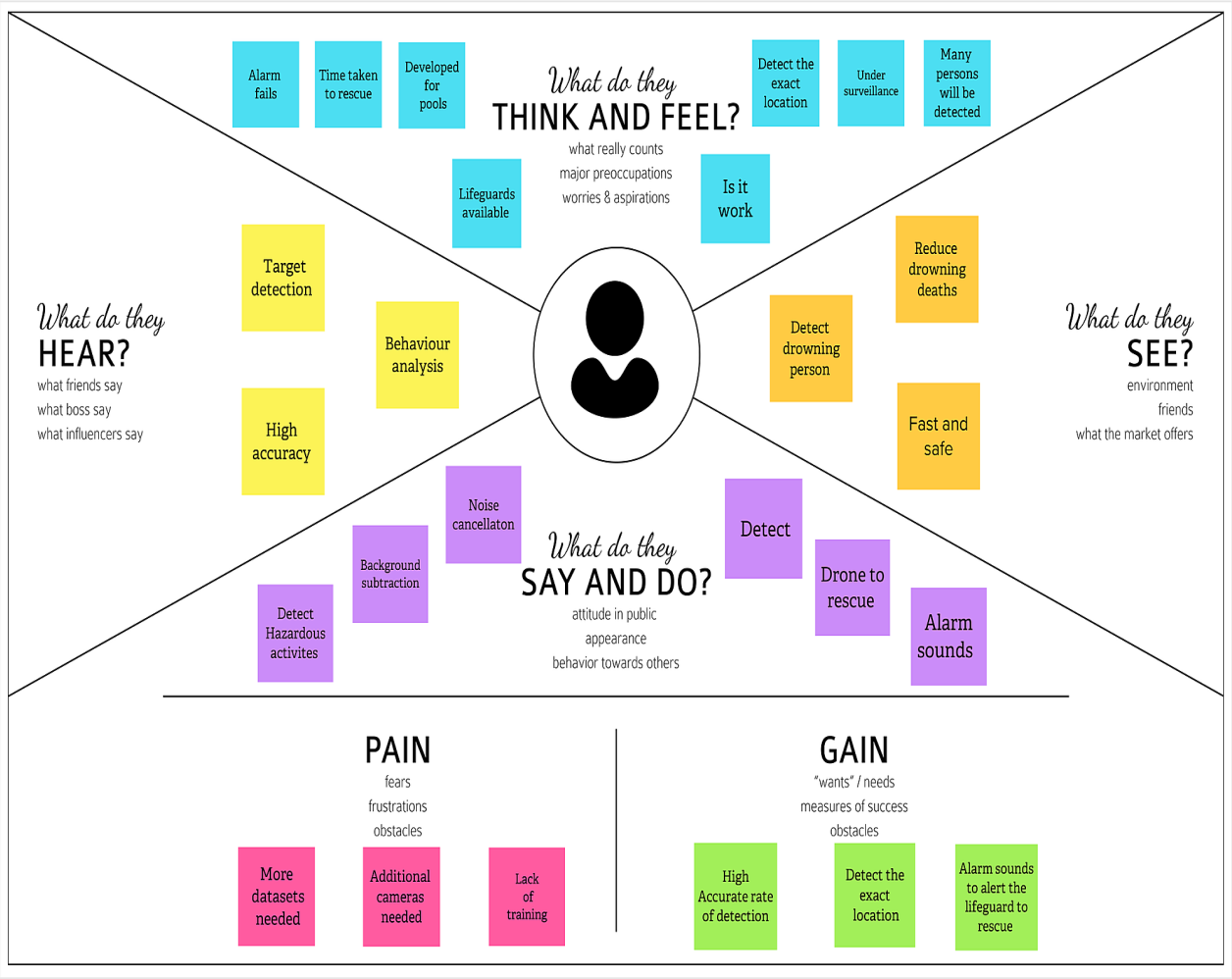


Fig 3.1.1 Empathy Map

3.2 IDEATION & BRAINSTORMING

Fig 3.2.1 Brainstorming and Idea Prioritization

3.3 PROPOSED SOLUTION

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	Abnormal behaviour or activity detection and prevent from drowning.
2.	Idea / Solution description	Our model using YOLO Model, we can predict drowning people in swimming pool. In specific, we use YOLO algorithm to get high accuracy results.
3.	Novelty / Uniqueness	Since we use YOLO algorithm, it takes less time to proceed, compare to other algorithm. The alarm will be triggered immediately right after the detection.
4.	Social Impact / Customer Satisfaction	Yearly 1.2 million people face unplanned death due to drowning globally. This dead rate will be reduced by implementing this solution.
5.	Business Model (Revenue Model)	Hence it is a lifesaving model, it can be used by beginners and irregular swimmers. It alerts near by swimmers who rescue the drowning one.
6.	Scalability of the Solution	YOLO model has a great flexibility architecture compare to others. Increase in depth scaling and width scaling and resolution scaling we can increase the scalability of this model. Real-time Object detection of yolo algorithm has high accuracy and fast.

3.4 PROBLEM SOLUTION FIT

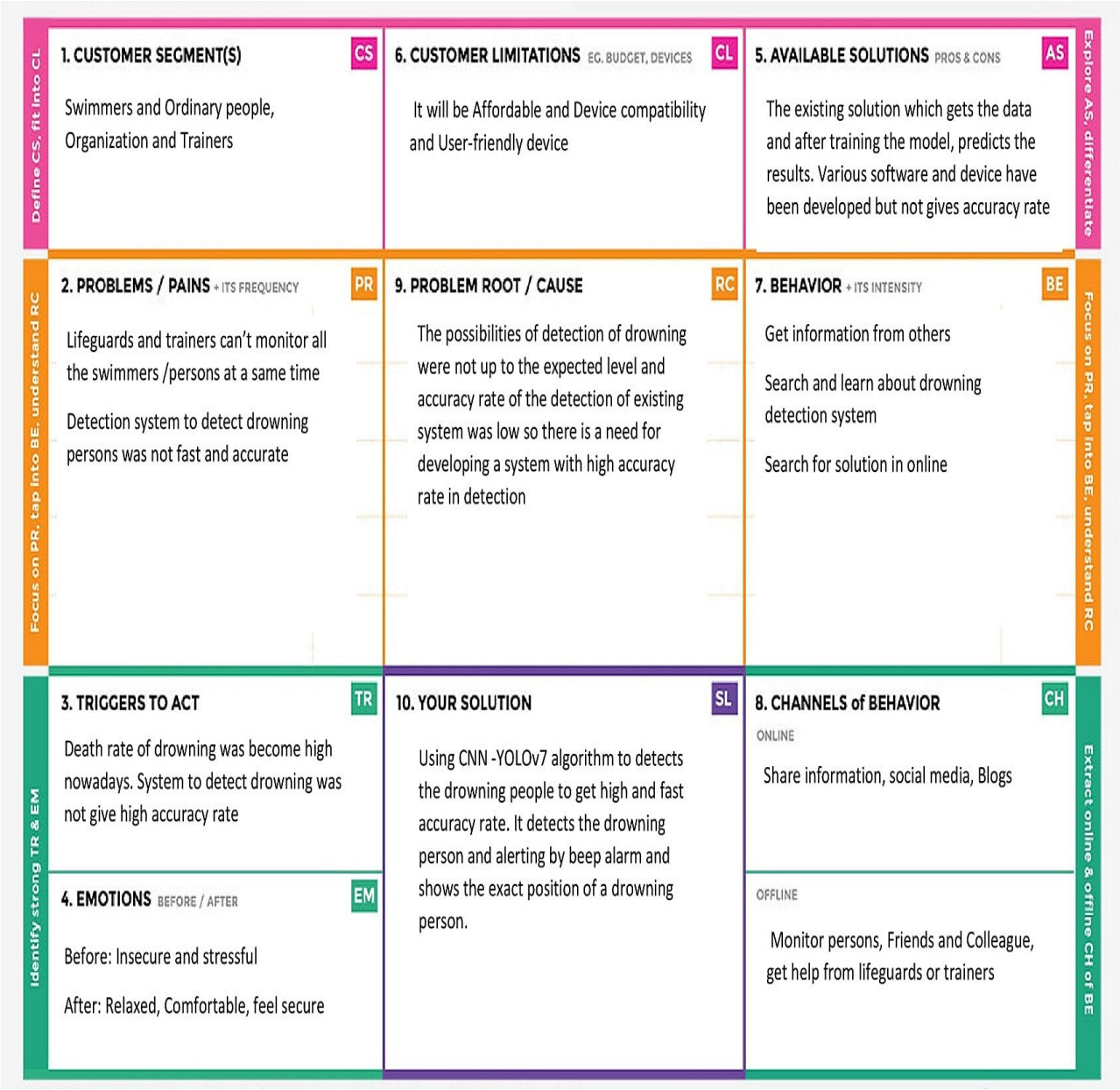


Fig 3.4.1 Problem Solution Fit

CHAPTER - 4
REQUIREMENT ANALYSIS

4.1 FUNCTIONAL REQUIREMENTS

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Registration	Registration Via Email
FR-2	User Confirmation	Create and store the data
FR-3	User Login	Login using Credentials
FR-4	Alarm system	Detect the drowning person Alert the lifeguard by trigger the alarm
FR-5	Output	Image detection Report generation

4.2 NON- FUNCTIONAL REQUIREMENTS

NFR No.	Non-Functional Requirement	Description
NFR-1	Usability	It detect drowning and this model is used for detection of drowning
NFR-2	Security	Observing each and every body movement of the swimmers.
NFR-3	Reliability	Suitable for all the swimming pools.
NFR-4	Performance	Life guard can visually access the developing situation within seconds of the event first occurring and initiate the rescue procedure when necessary.
NFR-5	Availability	Software Accessible all the time
NFR-6	Scalability	It helps to detect drowning with high accuracy and fast..

CHAPTER - 5
PROJECT DESIGN

5.1 DATA FLOW DIAGRAM

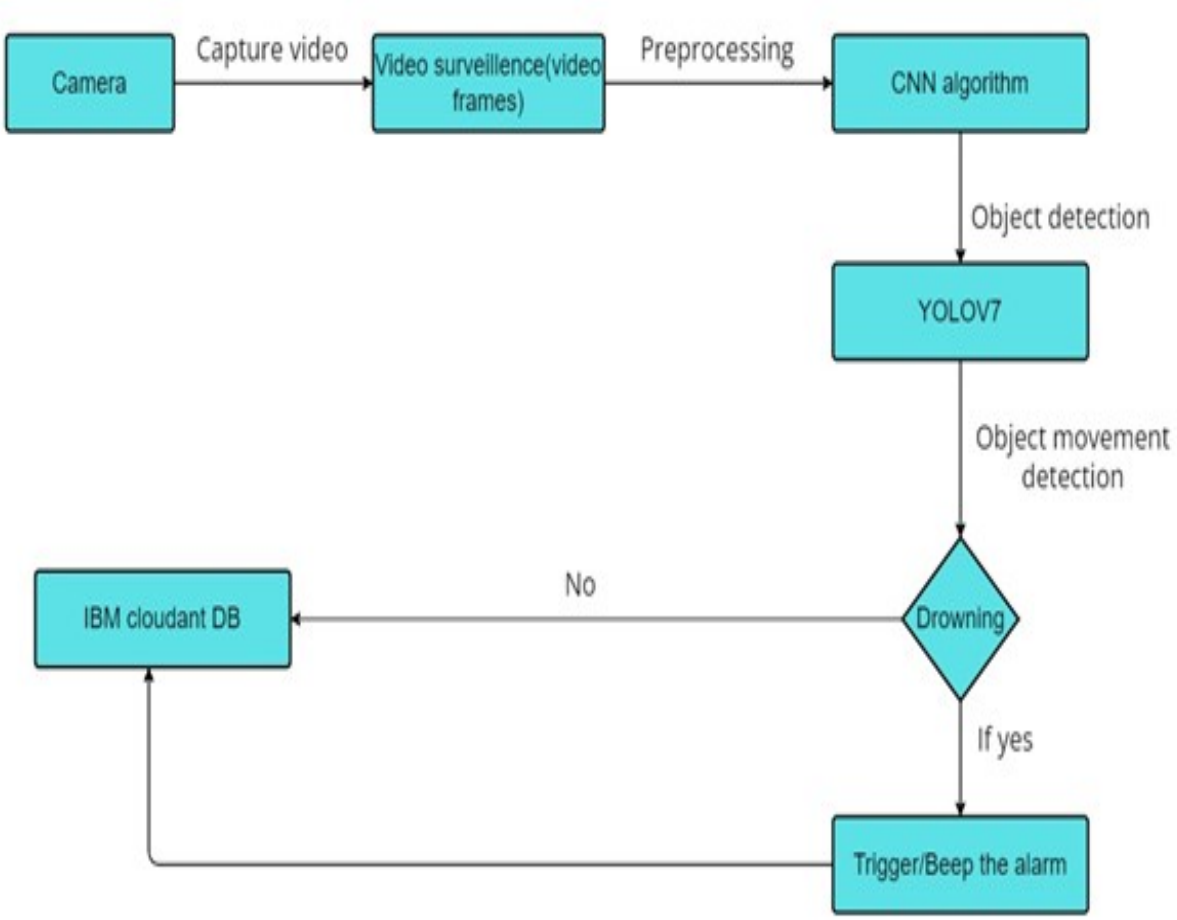


Fig 5.1.1 Dataflow Diagram

5.2 SOLUTION &TECHNICAL ARCHITECTURE

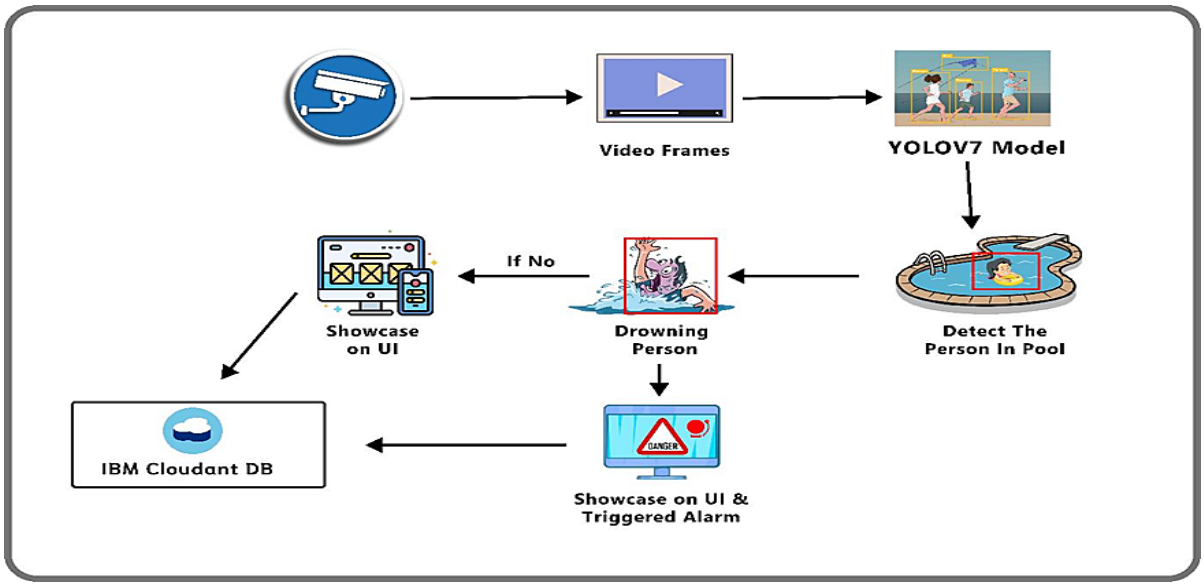


Fig 5.2.1 Solution Architecture

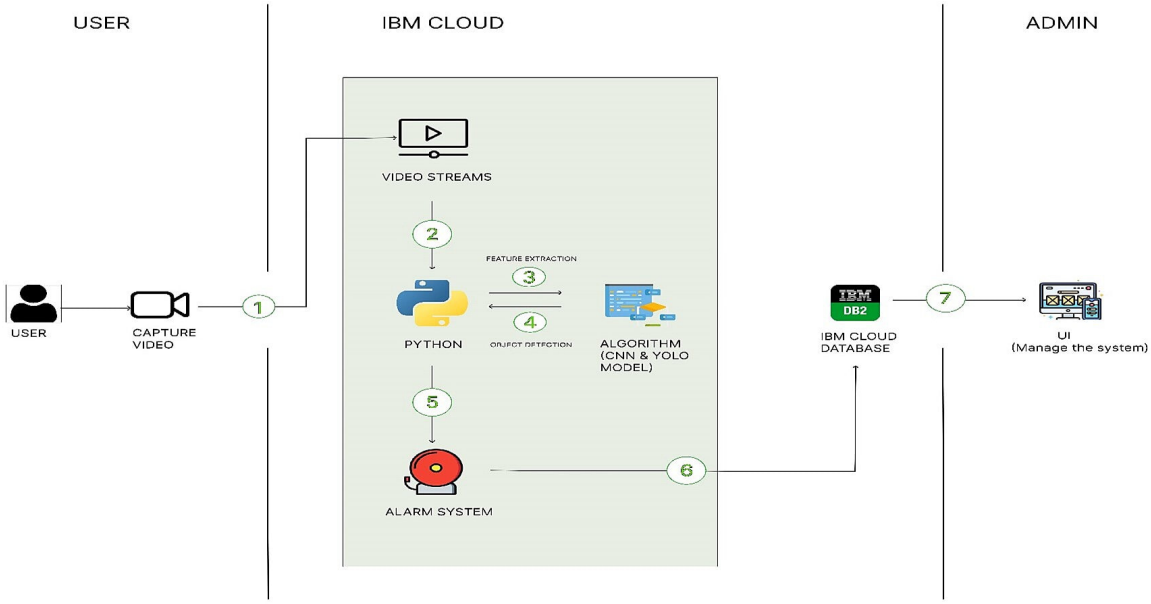


Fig 5.2.2 Technical Architecture

5.3 USER STORIES

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer (Web User)	Register	USN-1	Register into the application	User can access the account.	Medium	Sprint-1
	Login	USN-2	Login into the application by using login Credentials	User can login into the application.	Medium	Sprint-1
		USN-3	Stores the database	Storage the database	High	Sprint-2
		USN-4	Logout from the application	logout from application	Medium	Sprint-4
Admin	Pre-processing	USN-5	Train and test the model	Train the model by using datasets	High	Sprint-1
	Detection of drowning	USN-6	Detection the person by using trained model	Detection	High	Sprint -2

		USN-7	Swimmers can be detected through their actions	Detection of drowning	High	Sprint-3
	Alarm rings	USN -8	Alarm rings When the system detects drowning person	Alarm Beeps	High	Sprint-4

CHAPTER-6
PROJECT PLANNING & SCHEDULING

6.1 SPRINT PLANNING & ESTIMATION

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Registration	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password.	3	High	Jothika S
Sprint-1	Login	USN-2	As a user, I can log into the application by entering email & password.	3	High	Kavitha R
Sprint-1		USN-3	In prediction page, as a user, I can upload the data for detect the drowning person.	3	Medium	Kaviya K
Sprint-1	Dataset collection	USN-4	We can collect number of datasets; we can get high accuracy depends on collecting the number of	3	High	Ajitha R

			datasets.			
Sprint-2	Data Pre-processing	USN-5	The dataset is extracted and is used to train the model.	3	High	Ajitha R
Sprint-2	Train the model	USN-6	Train the model.	5	High	Jothika S
Sprint-2		USN-7	Test the model.	8	High	Kaviya K
Sprint-3	Detection	USN-8	Load the trained model.	5	High	Ajitha R
Sprint-3		USN-9	Now the real-time data to classify it by using a trained model to predict the output of the given real-time input.	8	High	Jothika S
Sprint-4		USN-10	If in case the person is drowning, the system will ring an alarm to notify for rescue the person.	8	High	Kavitha R
Sprint-4	Logout	USN-11	As a user, I can detect and logout from the application.	3	Medium	Kaviya K

6.2 SPRINT DELIVERY SCHEDULE

Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date (Actual)
Sprint-1	12	6 Days	24 Oct 2022	29 Oct 2022	12	01 Nov 2022
Sprint-2	16	6 Days	31 Oct 2022	05 Nov 2022	16	06 Nov 2022
Sprint-3	13	6 Days	07 Nov 2022	12 Nov 2022	13	13 Nov 2022
Sprint-4	11	6 Days	14 Nov 2022	19 Nov 2022	11	19 Nov 2022

6.3 REPORTS FROM JIRA

SPRINT-1

Fig 6.3.1 Sprint - 1

SPRINT-2

Fig 6.3.2 Sprint - 2

SPRINT-3

Fig 6.3.3 Sprint - 3

SPRINT-4

Fig 6.3.4 Sprint - 4

CHAPTER-7
CODING & SOLUTIONING

7.1 FEATURE 1

Object Detection :

By efficiently seeing through the water and enabling the system to track swimmers to determine whether they are drowning, it aids in the analysis of the videos taken by the cameras.Here, the swimmer is identified, image features are retrieved, and image processing is done to aid in finding the drowning person.

Fig 7.1.1 Coding & Solutioning

7.2 FEATURE 2

Alarm System :

This proposed system beeps alarm when person is drowning.Detect the object by using this algorithm accurately and fast.

Fig 7.2.1 Coding & Solutioning

CHAPTER-8
TESTING

8.1 TEST CASES

Test case ID	Component	Test Scenario	Steps To Execute	Expected Result	Test Data	Actual result	Status
LoginPage_TC_OO1	Index Page	Verify user is able to see the Login page	1.Click URL and go 2.Click on Login option on the navigation bar 3. Verify login page displayed or not	Login page should display	http://127.0.0.1:5000/	Working as expected	Pass
LoginPage_TC_OO2	Login page	Verify the UI elements in Login	UI elements: a. email text box b. password text box c. Login button d. new customer? Create account link e. Last password? Forgot password	Application should show below UI elements: a. email text box password text box c. Login button with orange colour d. new customer? Create account link e. Last password? Forgot password	http://127.0.0.1:5000/login	Working as expected	Pass

LoginPage_TC_OO3	Login page	Verify user is able to log into application with Valid credentials	1.Enter URL and click go 2.Click on Login option on the navigation bar 3. Enter Valid email in email text box 4.Enter valid password in password text box 5.Click on login button	Login into the prediction page	Email: 2k19cse005@kiot.ac.in password: Ajitha@52	Working as expected	Pass
LoginPage_TC_004	Login page	Verify user is able to log into application with Invalid credentials	1.Enter URL and click go 2. Click on Login option on the navigation bar 3.Enter Invalid password or email 5.Click on login button	Application should show 'Invalid email or password ' validation message.	Email: 2k19cse005@kiot.ac.in password: Ajitha#12	Working as expected	Pass
RegisterPage_TC_OO5	Index Page	Verify user is able to see the Register page	1.Click URL and go 2.Click on Register option on the navigation bar 3. Verify Register page displayed or not	Register page displayed	http://127.0.0.1:5000/register	Working as expected	Pass

RegisterPage_TC_OO6	Register page	Verify the UI elements in Register	UI elements: a. email text box b. password text box c. Username d. Register button e. already have an account? Login	Application should show below UI elements: a. email text box b. password text box c. Username d. Register button e. already have an account? Login	http://127.0.0.1:5000/register	Working as expected	Pass
RegisterPage_TC_OO7	Register page	If user don't have an account? Register User has able to register	1.Enter URL and click go 2.Click on Register option on the navigation bar 3. Enter Username 4. Enter Valid email in email text box 5.Enter valid password in password text box 6. Re-enter the password for confirmation 7.Click on Register button	Application should show 'Registration Successfully please login using your details' validation message.	Email: 2k19cse005@kiot.ac.in password: Ajitha@52	Working as expected	Pass

RegisterPage_TC_008	Register page	If user have an account?	1.Enter URL and click go 2.Click on Register option on the navigation bar 3. Enter Username 4. Enter Valid email in email text box 5.Enter valid password in password text box 6. Re-enter the password for confirmation 7.Click on Register button	Application should show ' Already a member please login using your details' validation message.	Email: 2k19cse005@kiot.ac.in password: Ajitha#12	Working as expected	Pass
PredictionPage_TC_O 10	Prediction Page	Verify File upload for detection	1.Click URL and go 2.Click on Register option on the navigation bar 3. After Login into application 4.choose a file for detection in prediction page	File upload for detection	http://127.0.0.1:5000/prediction	Working as expected	Pass

LogoutPage_TC_O11	Prediction Page	Verify user is able to see the Prediction page	1.Click URL and go 2.Click on Register option on the navigation bar 3. After Login into application 4. prediction page displayed 5. In prediction page, click the logout option in navigation bar 6. Verify logout page displayed or not	Logout page displayed	http://127.0.0.1:5000/logout	Working as expected	Pass

8.2 USER ACCEPTANCE TESTING

1. Purpose of Document

The purpose of this document is to briefly explain the test coverage and open issues of the [ProductName] project at thetime of the release to User Acceptance Testing (UAT).

2. Defect Analysis

This report shows the number of resolved or closed bugs at each severity level, and how they were resolved

Resolution	Severity 1	Severity 2	Severity 3	Severity 4	Subtotal
By Design	7	4	1	2	14
Duplicate	1	0	3	0	4
External	2	2	0	1	5
Fixed	8	2	2	10	22
Not Reproduced	0	0	1	0	1
Skipped	0	0	1	1	2
Won't Fix	0	3	2	1	6
Totals	18	11	10	15	54

3. Test case Analysis

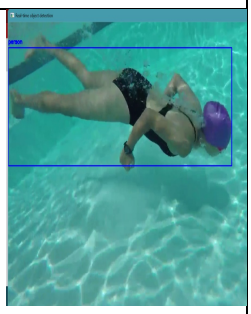
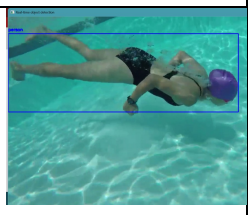
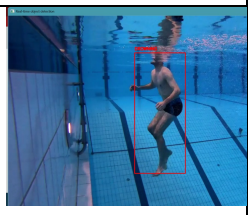
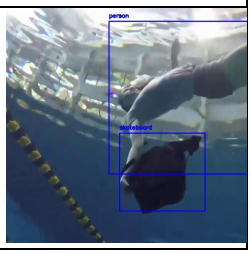
This report shows the number of test cases that have passed, failed, and untested

Section	Total Cases	Not Tested	Fail	Pass
Print Engine	11	0	0	11
Client Application	48	0	0	48
Security	1	0	0	1
Outsource Shipping	3	0	0	3
Exception Reporting	9	0	0	9
Final Report Output	4	0	0	4
Version Control	2	0	0	2

CHAPTER-9

RESULTS

9.1 PERFORMANCE METRICS

S.No.	Parameter	Values	Screenshot
1.	Model Summary	YOLO (You Only Look Once) is a method / way to do object detection. It is the algorithm /strategy behind how the code is going to detect objects in the image. The official implementation of this idea is available through DarkNet (neural net implementation from the ground up in C from the author).	
2.	Accuracy	Training Accuracy - 75% Validation Accuracy -73%	
	Accuracy Score	Drowning Detected- NO Accuracy Score - 75.8%	
	Accuracy Score	Drowning Detected- YES Accuracy Score - 85%	
3.	Confidence Score (Only Yolo Projects)	Class Detected - 70% Confidence Score - 78.3%	

CHAPTER-10

ADVANTAGES & DISADVANTAGES

ADVANTAGES

- The system's unique capacity to track swimmers and monitor them allows it to reduce and prevent drowning-related fatalities.
- The detection accuracy of the proposed system is high.
- Fast detection of drowning person.

DISADVANTAGES

- This system needs constant observation.
- Detection of person in low light or night time.

CHAPTER-11
CONCLUSION

In conclusion, this study has demonstrated that the visual processing skills of people with experience of lifeguarding and lifesaving skills can be assessed in a simple computer test. The core findings of these experiments are clear, with lifeguards detecting drowning swimmers faster and more frequently than non-lifeguards. There was some evidence that small gains in time to first fixate the drowning swimmers and the processing time may add up to make the significantly faster response times of lifeguards in detection of drowning swimmers.

CHAPTER-12
FUTURE SCOPE

The system could benefit from having an additional set of cameras to recognise and confirm a drowning or a dangerous behaviour on the premises in the future for convenience's sake.The collection of a night time dataset that boosts the precision of the data in low light could help both drown and hazardous activity identification.

CHAPTER-13
APPENDIX

13.1 SOURCE CODE

```
import cvlib as cv
from cvlib.object_detection import draw_bbox
import cv2 , time
import numpy as np
from playsound import playsound
from flask import Flask , render_template , request,redirect
from cloudant.client import Cloudant
client = Cloudant.iam('83752889-cd2c-4ae9-8a1c-98286fc2dc8f-
bluemix','FHh8N5uZqElNN8QpfHgys93ANfx_-cX_Eytq6WkULDEo',
connect=True)
app=Flask(__name__)
my_database = client.create_database('my_database')
app=Flask(__name__)
@app.route('/')
def index():
    return render_template('index.html')
@app.route('/index.html')
def home():
    return render_template("index.html")
@app.route('/register')
def register():
    return render_template('register.html')
@app.route('/afterreg', methods=['POST'])
def afterreg():
    x = [x for x in request.form.values()]
```

```
  print(x)
   data = {
      'email': x[1],  # Setting _id is optional
      'username': x[0],
      'password': x[2]
   }
   print(data)
   query = {'email': {'$eq': data['email']}}
   docs = my_database.get_query_result(query)
   print(docs)
   print(len(docs.all()))
   if (len(docs.all()) == 0):
      url = my_database.create_document(data)
      return render_template('register.html', pred="Registration Successful,
please login using your details")
   else:
      return render_template('register.html', pred="You are already a member,
please login using your details")
@app.route('/login')
def login():
   return render_template('Login.html')
@app.route('/afterlogin', methods=['POST'])
def afterlogin():
   user = request.form['email']
   passw = request.form['password']
   query = {'email': {'$eq': user}}
   docs = my_database.get_query_result(query)
   if (len(docs.all()) == 0):
      return render_template('Login.html', pred="The username is not found.")
```

```
    else:
        if ((user == docs[0][0]['email'] and passw == docs[0][0]['password'])):
            return redirect("/prediction")
        else:
            return render_template('Login.html', pred="Invalid email or password")
@app.route("/demo")
def demo():
    return render_template('demo.html')
@app.route('/logout')
def logout():
    return render_template('Logout.html')
@app.route("/prediction" , methods = ['GET' , 'POST'])
def predict():
    if request.method == 'POST':
        image = request.files['file']
        webcam = cv2.VideoCapture(image.filename)
        t0 = time.time()
        centre0 = np.zeros(2)
        isDrowning = False
        while True:
            status, frame = webcam.read()
            if not status:
                print("Could not read frame")
                exit()
            bbox, label, conf = cv.detect_common_objects(frame)
            if(len(bbox)>0):
                    bbox0 = bbox[0]
                    centre = [0,0]
                    centre =[(bbox0[0]+bbox0[2])/2,(bbox0[1]+bbox0[3])/2 ]
                    hmov = abs(centre[0]-centre0[0])
```

```
                vmov = abs(centre[1]-centre0[1])
            x=time.time()
                threshold = 10
                if(hmov>threshold or vmov>threshold):
                    print(x-t0, 's')
                    t0 = time.time()
                    isDrowning = False
                else:
                    print(x-t0, 's')
                    if((time.time() - t0) > 10):
                        isDrowning = True
                print('bbox: ', bbox, 'centre:', centre, 'centre0:', centre0)
                print('Is he drowning: ', isDrowning)
                centre0 = centre
        out = draw_bbox(frame, bbox, label, conf,isDrowning)
        cv2.imwrite('image.jpg',out)
        if isDrowning:
            playsound('alarm.mp3')
        cv2.imshow("Real-time object detection", out)
        if cv2.waitKey(1) & 0xFF == ord('q'):
            break
    webcam.release()
    cv2.destroyAllWindows()
  return render_template('prediction.html')
if __name__=="__main__":
  app.run(debug=True)
```

13.2 GITHUB & PROJECT DEMO LINK

GITHUB LINK:

https://github.com/IBM-EPBL/IBM-Project-18772-1659689839

PROJECT DEMO LINK:

https://youtu.be/pG8yFDtJQgg

CHAPTER-14
REFERENCES

[1] A. I. N. Alshbatat, "Automated Vision-based Surveillance System to Detect Drowning Incidents in Swimming Pools.," in In Proceedings of the 2020 Advances in Science and Engineering Technology International Conferences (ASET), Dubai, UAE, 2020.

[2] H. M. A., "The Swimmers Motion Detection Using Improved VIBE Algorithm," in International Conference on Robotics and Automation in Industry (ICRAI), 2019.

[3] E. How-Lung, "An automatic drowning detection surveillance system for challenging outdoor pool environments"

[4] Ajil R., Srinivasan, K.: A novel drowning detection method for safety of swimmers. In: 2018 20th National Power Systems Conference (NPSC), pp. 1–6. IEEE (2018)

[5] SwimEye, "A drowning detection system for any pool," Swimeye.com, [Online]. Available: https://swimeye.com/swimeye/. [Accessed 25 November 2020].

[6] L. Wenmiao, "A vision-based approach to early detection of drowning incidents in swimming pools," in IEEE transactions on circuits and systems for video technology 14.2 159-178., 2004.

[7] Ajil R., Srinivasan, K.: A novel drowning detection method for safety of swimmers. In: 2018 20th National Power Systems Conference (NPSC), pp. 1–6. IEEE (2018).

[8] Wang, F., Ai, Y., Zhang, W.: Detection of early dangerous state in deep

water of indoor swimming pool based on surveillance video (2021).