VIRTUALEYE - LIFE GUARD FOR SWIMMING POOLS TO DETECT ACTIVE DROWNING

PROJECT REPORT

HX5001

NALAIYA THIRAN PROJECT BASED LEARNING ON PROFESSIONAL READINESS FOR INNOVATION, EMPLOYMENT AND ENTERPRENEURSHIP

Submitted By

TEAM ID: PNT2022TMID35895

PRAJEEN R G   2019504560
HARI @ ARJUN K   2019504525
HARISH G   2019504026
SHWETHA L S   2019504586

Under the Guidance of

INDUSTRY MENTOR   : SWATHI
FACULTY MENTOR   : DR. V. SATHIESH KUMAR
FACULTY EVALUATOR : DR. K. MARIAMMAL

NOVEMBER 2022

Table of Contents

ABSTRACT

The death from drowning has caused one third of untimely death in the world. This happens to small children and newbie swimmers to prevent this from happening there are various safety measures taken to prevent drowning in swimming pool. 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. This paper compares the detection accuracy of different detection algorithms and analyzes the detection effect of different pool angles and different swimmer densities.Designed a user-friendly website for virtualeye. Implemented a clouded based authentication system using CloudantDB. Implemented login and user registration flow for the app. Developed a program to detect people using YOLOv4 and alarm if they are drowning for 10 seconds. Coded the program to work on the pre-recorded video clip to show as proof of concept.Swimming is one of the best exercises that helps people to reduce stress in this urban lifestyle.

Swimming pools are found larger in number in hotels, and weekend tourist spots, and barely people have them in their houses and backyards. Beginners, especially, often feel it difficult to breathe underwater which causes breathing trouble which in turn causes a drowning accident. Worldwide, drowning produces a higher rate of mortality without causing injury to children. Children under six of their age are found to be suffering the highest drowning mortality rates worldwide. Such kinds of deaths account for the third cause of unplanned death globally, with about 1.2 million cases yearly.

To overcome this conflict, a meticulous system is to be implemented along the swimming pools to save human life. Once we have the working drowning detection model we can feed live video footage of the swimming pool to it so that it can keep detecting continuously for any drowning activities. If drowning is detected it will be highlighted on the system screen as well as alarms will be raised to alert security guards so that they can initiate a rescue. Availability of better datasets, modern methodologies, and technologies with high computational power accompanied by high-quality surveillance cameras, will help to improve the accuracy of drowning detection & even can be used in adverse conditions. After the implementation of all these essentials, this system also can be used on sea beaches for drowning detection

1. INTRODUCTION

1.1. PURPOSE OF THE PROJECT

Swimming is one of the best exercises that helps people to reduce stress in this urban lifestyle. Beginners, especially, often feel it difficult to breathe underwater which causes breathing trouble which in turn causes a drowning accident. Worldwide, drowning produces a higher rate of mortality without causing injury to children. Children under six of their age are found to be suffering the highest drowning mortality rates worldwide. Such kinds of deaths account for the third cause of unplanned death globally, with about 1.2 million cases yearly. To overcome this conflict, a meticulous system is to be implemented along the swimming pools to save human life.

1.2. PROJECT OVERVIEW

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.

2. LITERATURE SURVEY

2.1. EXISTING PROBLEM

Safety is paramount in all swimming pools. The current systems expected to address the problem of ensuring safety at swimming pools have significant problems due to their technical aspects such as underwater camera and methodological aspects such as the need or human intervention in the rescue mission. The automated visual-based monitoring system can help to reduce drownings and assure pool safety effectively. In order to quickly help life savers judge whether people are drowning in the swimming pool. The VirtualEye software works in close integration with the cameras installed in the pool to continuously to scan the swimming pool. The First, by analyzing the spatial distribution of swimming pool when swimmers are normally swimming, the data labeling and swimmer detect methods are determined. Second, a behavior recognition framework of swimmers on the basis of YOLOv4 algorithm (BRYOLOv4) is proposed. The spatial relationship between the location information of the target and swimming/drowning area of swimming pool is

analyzed to determine the swimmer's drowning or swimming behavior. It introduces a revolutionary technology that identifies drowning victims in a minimum amount of time and dispatches an automated drone to save them. Using convolutional neural network (CNN) models, it can detect a drowning person in three stages. Whenever such a situation like this is detected, the inflatable tube-mounted self-driven drone will go on a rescue mission, sounding an alarm to inform the nearby lifeguards. The system also keeps an eye out for potentially dangerous actions that could result in drowning. This system's ability to save a drowning victim in under a minute has been demonstrated in prototype experiments' performance evaluations. The live video stream from our underwater cameras is automatically monitored by our "state-of-the-art" object recognition software. When it detects a swimmer in distress on the bottom of the pool, it will raise a radio alarm to pool lifeguards and an visual alarm to our Monitoring & Control Station. Lifeguards can visually assess the developing situation within seconds of the event first occurring, and initiate their rescue procedure when necessary.

2.2. REFERENCES

[1] Upulie Handalage, Nisansali Nikapotha, Chanaka Subasinghe, Tereen Prasanga "Computer Vision Enabled Drowning Detection System" - 2021

Video-based systems and wearable sensor-based systems are two types of existing drowning detection technologies. It will use Object detection using different techniques and will usage of Convolutional Neural Network (CNN) architecture in Deep Neural Networks (DNNs) has added a significant shift in learning more complicated, informative characteristics in images as compared to older techniques. Then, Drowning Detection and Tracking to avoid drowning events utilizing an alert system. Activity Detection using Computer Vision has Current work on human motion prediction has been focused on two independent but complementary subtasks, according to Anand Gopalkrishnan, Short-term motion prediction, which is quantitatively evaluated by measuring the mean squared error (MSE) over a short period, and long-term motion prediction, qualitatively evaluated by visual inspections of samples over a long period. Short-term models would be valuable in motion tracking applications because these jobs are applicable in several domains of work. On the other hand, long-term models might be valuable for creating computer graphic tools due to their broad applicability. Additionally, both models could be useful in human gait analysis, kinematics research, and human-computer interaction.

ADVANTAGES:
- The monitoring system can help to reduce drowning and assure pool safety effectively.
- This system ability to save a drowning victim in under a minute has been demonstrated in prototype experiments.

DISADVANTAGES:
- 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 wearable-based system is the discomfort of use, which may lead to younger children seeking to alleviate the discomfort by removing the device, which is an unsubstantiated theory.

[2] Abdel Ilah N. Alshbatat, Shamma Alhameli, Shamsa Almazrouei, Salama Alhameli, Wadhha Almarar "Automated Vision-based Surveillance System to Detect Drowning Incidents in Swimming Pools " - 2020

In recent years, there has been an interest in integrating computer vision in swimming pool surveillance systems. Automating such a process will provide the communities with an efficient way of detecting drowning incidents that may occur while swimming. a hybrid system that will automatically detect a drowning person and then set off an alarm to alert the lifeguards has been developed. The system mainly consists of three modules: a vision module, an event-inference module, and an event-driven module. The vision module is responsible for monitoring and detecting the position of the person who is drowning. The event inference module isresponsible for determining a swimmer's position, velocity, and path of the movement. The event-driven module is responsible for initiating the rescue by sending an alarm alerting the lifeguard. The main contribution of this project is to develop a system for monitoring swimming pool to prevent the onset of a drowning incident.

ADVANTAGES:
- This system don't have to wait until life guard comes to rescue because it has uplifting mesh.
- This is very fast process.
- More effective and cost Efficient than previous other models.

DISADVANTAGES:

- Internet connection is necessary to use GPS or sending alert messages. Sometimes to send messages SIM balance may be required.

[3] Muhammad Aftab Hayat, Goutian Yang, Atif Iqbal, Adeel Saleem, Adil hussain, Muhammad Mateen "The Swimmers Motion Detection Using Improved VIBE Algorithm " - 2019

This paper proposed a novel method for drowning person detection in the swimming pool using video images. For background extraction and to update the exact motion n area from the whole video using frame by frame difference vibe algorithm is used. Static and dynamic features are detected to recognize the normal swimmer and drowning person. The present invention discloses videobased swimming pools drowning event detection method. In the detection process Time of map(Tom), the method is used to improve the traditional VIBE result. The sequence of video images of the swimming pool is collected in real-time by using a camera installed above the water surface, which mainly includes three steps of swimmers detection, swimmers tracking and drowning person behavior analysis. In the aspect of swimmer detection, an improved VIBE swimmer detection algorithm is proposed, and the algorithm is used to determine the swimmer's position. The swimmer tracking and particle filter based on the color distribution model which is combined with the nearest neighbor data association algorithm to achieve tracking of multiple swimmers. In the analysis of drowning behavior, three characteristics of drowning behavior are proposed to determine whether the swimmer is drowning. The invention can monitor the swimming pool in real-time through the camera installed above the water surface in a real public swimming place, and automatically detect the drowning person, which has great engineering application value.

ADVANTAGES:
- The full security system promotes the development of water lifesaving services, which is also the objective requirement for the current development of swimming lifeguards.

DISADVANTAGES:
- The disadvantage of this management method is that it cannot be entered into the pool area all the time. The management of the refinement, reliability, and reaction speed is very slow because the processing capability of the information about the swimmer is very weak. It is difficult to effectively protect public safety in the venue.

[4] Ajil Roy, Dr. K. Srinivasan "A novel drowning detection method for safety of swimmers " - 2018

Effective drowning detection methods are essential for the safety of swimmers. In this paper, a novel type of drowning detection method addressing many limitations of prevailing drowning detectors is proposed. The proposed method ensures detection of drowning and reporting at the earlier stages. The proposed drowning detection method is also a generic solution that suites different water bodies from pools to oceans, and an economically viable method useful for both low and middle income countries. The prototype of the drowning detection method is developed and demonstrated and model of the system is simulated in Proteus design suite. The results of the simulation and hardware experimentation are also reported.

ADVANTAGES:
- This will help the lifeguard to search for his previous medical records as does the patient had any heart or lungs diseases etc. This information will provide an additional advantage while doing the rescue operation and while doing first aid.

DISADVANTAGES:
- It is also a reliable solution where the life guards have difficulty to monitor the swimmers like a highly crowded sea.

[5] Nasrin Salehi and Maryam Keyvanara, Seyed Amirhassan Monadjemmi "An Automatic Video-based Drowning Detection System for Swimming Pools Using Active Contours " - 2016

Safety in swimming pools is a crucial issue. In this paper, a real time drowning detection method based on HSV color space analysis is presented which uses prior knowledge of the video sequences to set the best values for the color channels. Our method uses a HSV thresholding mechanism along with Contour detection to detect the region of interest in each frame of video sequences. The presented software can detect drowning person in indoor swimming pools and sends an alarm to the lifeguard rescues if the previously detected person is missing for a specific amount of time. The presented algorithm for this system is tested on several video sequences recorded in swimming pools in real conditions and the results are of high accuracy with a high capability of tracking individuals in real time. According to the evaluation results, the number of false alarms generated by the system is minimal and the maximum alarm delay reported by the system is 2.6 sec which can relatively be reliable compared to the acceptable time for rescue and resuscitation.

ADVANTAGES:
- Video surveillance can be used as a tool for monitoring and security.
- Our algorithm was able to detect all the drowning conditions along with the exact position of the drowning person in the swimming pool and had an average detection delay of 1.53 seconds, which is relatively low compared to the needed rescue time for a lifeguard operation.

DISADVANTAGES:
- Each year many people including children are drowned or very close to drowning in the deeps of the swimming pools, and the life guards are not trained well enough to handle these problems. This raises the need for having a system that will automatically detect the drowning person and alarm the lifeguards of such danger.

2.3. PROBLEM STATEMENT

Who:
An AI powered detector to be deployed at places where there is a high possibility of people swimming like pools, beaches, etc.

What:
Minimize the chances of people drowning in the water by recognizing the possible actions people make at the time of distress and panic and sending an alert to the control room for taking suitable measures to save them.

Barriers:
- Differentiation between swimming and drowning.
- Involves excessive investment for implementation.
- Timely alert and the accuracy of the same.

Cause:
The major factors that lead to lots of drowning accidents are careless lifeguards and the swimming pool authorities. Most times the carelessness can be attributed to the ambiguity of whether someone is drowning or not.

Emotion:
- Promotes swimming as a safe activity.
- Makes parents let their children go for a swim.
- Countless lives are saved more quickly than before.

3. IDEATION AND PROPOSED SOLUTION

3.1. EMPATHY MAP CANVAS
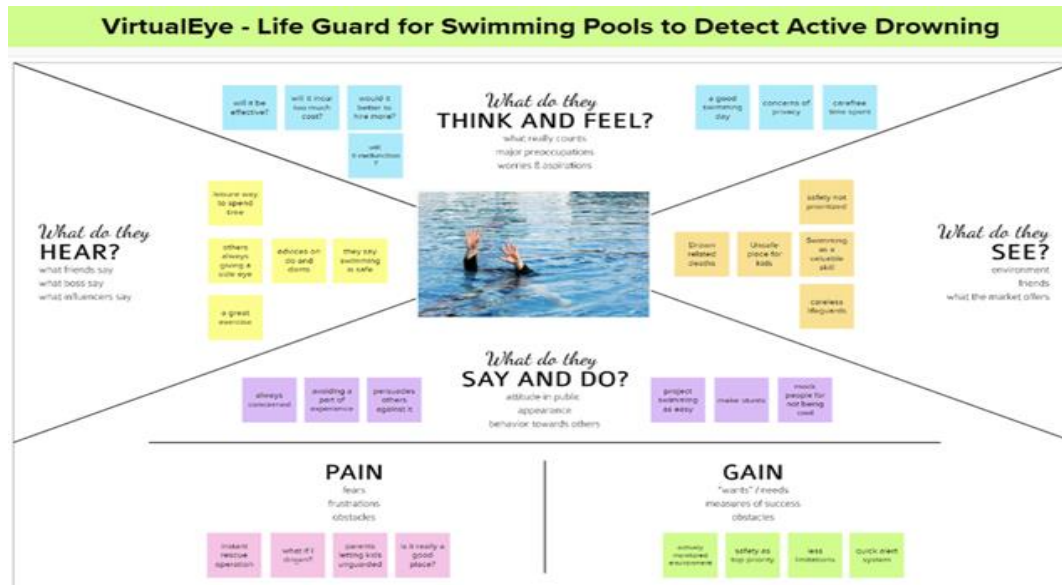
3.2. IDEATION AND BRAINSTROMING

3.3. PROPOSED SOLUTION

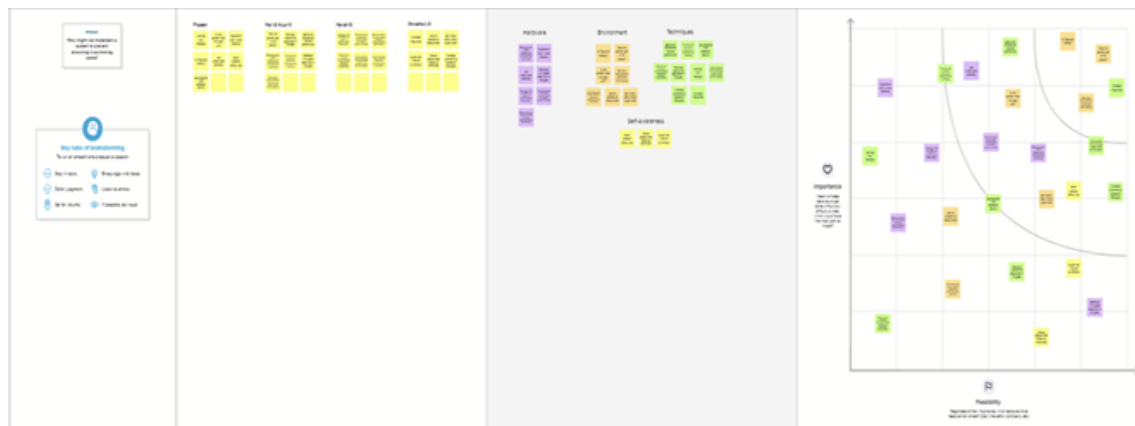| S.NO.	PARAMETER	DESCRIPTION
1.	Problem Statement	Swimming is one of the best exercises that helps people to reduce stress in this urban lifestyle. Swimming pools are found larger in number in hotels, and weekend tourist spots and barely people have them in their house backyard. Beginners, especially, often feel it difficult to breathe underwater which causes breathing trouble which in turn causes a drowning accident. Worldwide, drowning produces a higher rate of mortality without causing injury to children. Children under six of their age are found to be suffering the highest drowning mortality rates worldwide. Such kinds of deaths account for the third cause of unplanned death globally, with about 1.2 million cases yearly. To overcome this conflict, a meticulous 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 analysing 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 than an alert will be generated to attract lifeguards' attention.
2.	Idea / Solution description	To resolve the conflict, a system is being implemented along the swimming pools to save human lives. We are developing an underwater pool safety system that reduces the risk of drowning by studying body movement patterns and connecting cameras to artificial intelligence (AI) systems. Typically, such systems are created by installing more than 16 cameras underwater and on the ceiling and analysing the video feeds to detect any anomalies. However, as a point of contact, we use a single camera that streams video underwater and analyses swimmer positions to determine the likelihood of drowning; if the probability is higher, an alert is generated to draw the attention of lifeguards.
3.	Novelty / Uniqueness	1. Unlike other solutions, this is a human assistive system by nature. 2. Errors are minimised but the negligence of humans will not affect the working of this solution. 3. As the cameras are employed with ML, it can work 24/7 and even detect drowning of slightest amount. 4. Instead of using YOLOv1 or YOLOv2 this proposed solution incorporates the latest version of YOLO which is v3, this is way superior in performance and accuracy than the former.

3.4. PROBLEM SOLUTION FIT:

4. REQUIREMENT ANALYSIS
4.1. FUNCTIONAL REQUIREMENTS

Functional Requirements:

Following are the functional requirements of the proposed solution.

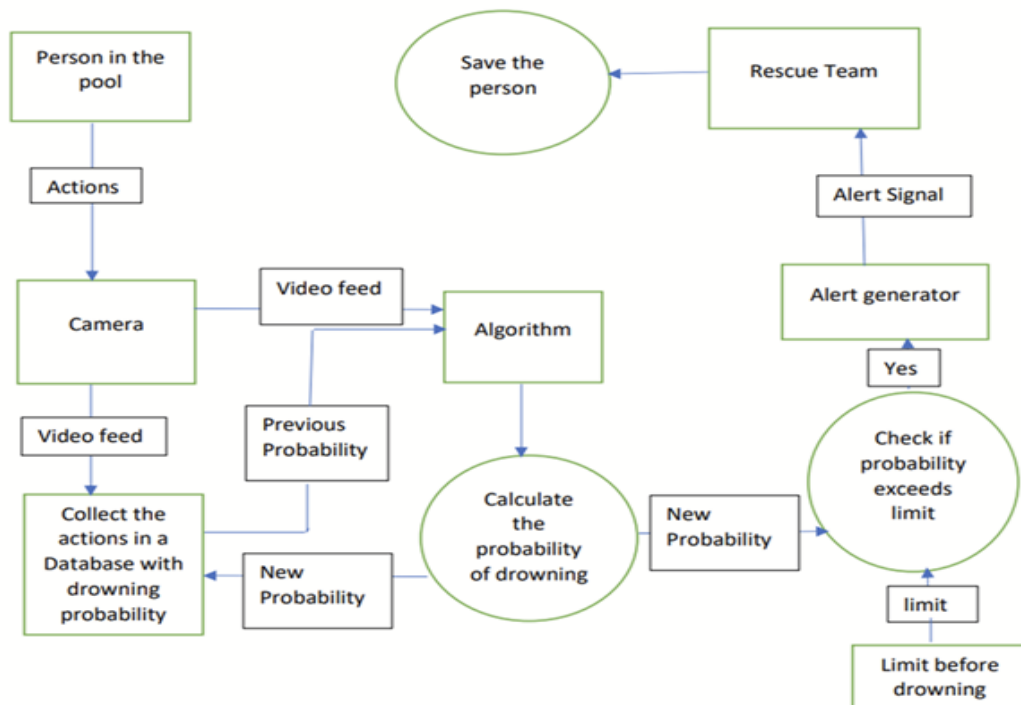| FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	Detection of People	Able to detect people in the water. Keep track of the count of people. Distinguish people from objects.
FR-2	Identify Danger	Identify people at risk of drowning. Distinguish drowning from other non-risky actions. Identify the actions people do to seek help.
FR-3	Probability of drowning	Calculate the degree of risk of drowning for every person. If the degree exceeds the limit, generate an alert.
FR-4	Communicate Danger	Let the rescue team know about the person drowning. Communicate the exact location of drowning.

4.2. NON- FUNCTIONAL REQUIREMENTS

Non-functional Requirements:

Following are the non-functional requirements of the proposed solution.

| FR No.	Non-Functional Requirement	Description
NFR-1	Usability	Easy to implement and requires minimal and readily available resources and components.
NFR-2	Security	Vigilant. An alert is sent to the lifeguard when the probability of drowning is assessed to be high.
NFR-3	Reliability	Highly reliable, the system can be developed by installing more cameras underwater to detect any anomalies.
NFR-4	Performance	The system provides promising results and any crisis can be prevented quickly.
NFR-5	Availability	Everyone is under the watchful eyes of the camera. The system works continuously to give reliable security services.
NFR-6	Scalability	Even a fully occupied pool has no effect on performance. The system performs optimally.

5. PROJECT DESIGN
5.1. DATA FLOW DIAGRAM

5.2. SOLUTION AND TECHNICAL ARCHITECTURE
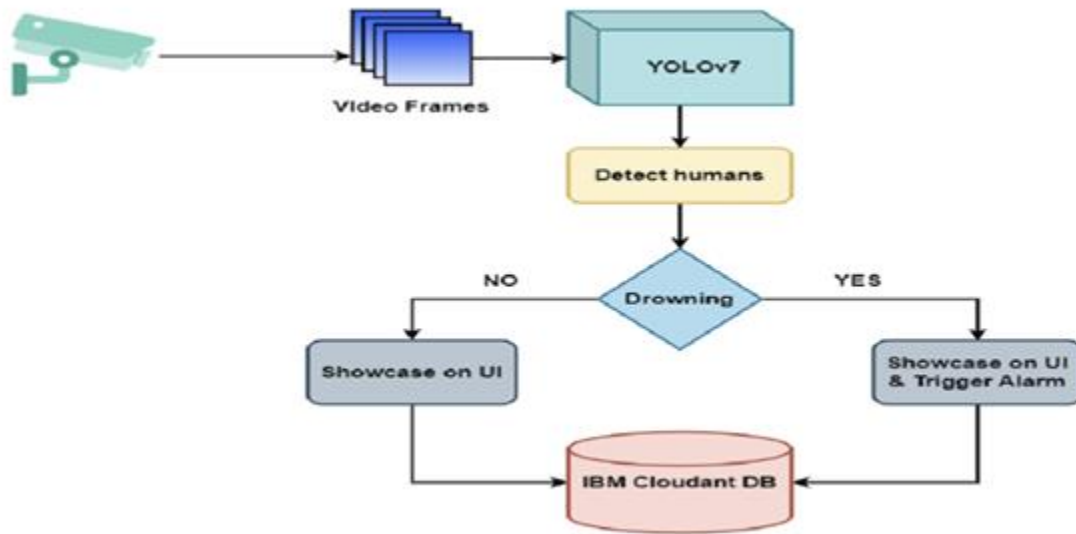
5.3. USER STORIES

User Stories :

Following are the user stories for the product.

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer (Mobile user/ Web user)	Sign in	USN-1	The user will open the browser and go to the URL bar.	I can type my URL.	High	Sprint-1
		USN-2	The user will type the project URL.	I can open my project URL.	High	Sprint-1
	Dashboard	USN-3	The user will see a permission popup.	I can give permission.	High	Sprint-1
		USN-4	The user will permit to access the camera.	I can allow the site to access the camera.	High	Sprint-1
Customer Care Executive	Maintenance	USN-5	The camera will be set in the proper position.	The camera angle will be placed properly.	High	Sprint-1
		USN-6	Checks whether the camera has a clear view.	The camera will have a proper view.	Medium	Sprint-2
		USN-7	Checks whether the lens of the camera is clean.	A clear picture is seen.	High	Sprint-2
		USN-8	Checks whether the camera has a good pixel quality	We will get a high-quality image.	Medium	Sprint-2
Administrator	Monitoring	USN-9	Sees the viewfinder.	We will get a focused image even if it is far.	High	Sprint-3
		USN-10	Reads and checks the manual.	Checks whether proper information is given.	Medium	Sprint-3
		USN-11	Checks whether a proper alert is given.	Gives out alert sounds.	High	Sprint-4

6. CODING AND SOLUTIONING

Feature 1:

Designed a user-friendly website for virtualeye.

Feature 2:

Implemented a clouded based authentication system using CloudantDB.

Feature 3:

Implemented login and user registration flow for the app.

Feature 4:

Developed a program to detect people using YOLOv4 and alarm if they are drowning for 10 seconds.

Feature 5:

Coded the program to work on the pre-recorded video clip to show as proof of concept.

6.1.DATABASE SCHEMA

FIELD NAME	FIELD TYPE	DESCRIPTION
_id	string	User email
Name	string	User full name
Psw	string	User password

7.TESTING

7.1. USER ACCEPTANCE TESTING

PURPOSE OF DOCUMENT:

This document serves as a quick reference for the Deep Learning Fundus Image Analysis for Early Detection of Diabetic Retinopathy project's test coverage and open issues as of the project's release for user acceptance testing.

DEFECT ANALYSIS:

This shows how many bugs were fixed or closed at each severity level and how they were fixed.

Resolution	Severity 1	Severity 2	Severity 3	Severity 4	Subtotal
By Design	5	4	2	3	14
Duplicate	1	0	3	0	4
External	2	3	0	1	6
Fixed	9	2	4	15	30
Not Reproduced	0	0	1	0	1
Skipped	0	0	1	1	2
Won't Fix	0	5	2	1	8
Totals	17	14	13	21	60

TEST-CASE ANALYSIS:
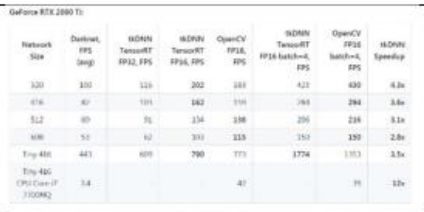
This report shows the number of test cases that have passed, failed, and untested.

Section	Total Cases	Not Tested	Fail	Pass
PrintEngine	9	0	0	9
ClientApplication	45	0	0	45
Security	2	0	0	2
OutsourceShipping	3	0	0	3
ExceptionReporting	9	0	0	9
FinalReportOutput	4	0	0	4
VersionControl	2	0	0	2

8. RESULTS

8.1. PERFORMANCE METRICS

S.No.	Parameter	Values	Screenshot
1.	Model Summary	-	
2.	Accuracy	Training Accuracy – 28% Validation Accuracy – 44%	
3.	Confidence Score (Only Yolo Projects)	Class Detected - Person Confidence Score – 87.23%	

9. ADVANTAGES AND DISADVANTAGES:

9.1. ADVANTAGES

- Drowning detection system that detects every dangerous situation and accident. The software works in close integration with the cameras installed in the pool to continuously scan the pool. Thanks to this combination of hardware, software, and profound innovations, the system would represent excellence in drowning detection.
- Features artificial intelligence technology that adapts to the needs of the user. It is the ultimate drowning detection system for those who demand ultimate safety.
- System would be able to record all the activities in the pools and classify critical situations from normal ones to keep track of what happened.
- Makes itself heard loud and clear in case of danger. The built-in notification system produces alarms within seconds on configurable devices. In addition, the technology would provide real-time location and images of the danger, making rescue operations easier.

- Identify swimmers in distress and raise an alarm which alerts the Lifeguards who can visually assess the developing situation in seconds the developing situation in seconds
- Can provide very stable monitoring and highly effective drowning incident detection.
- Suitable for any swimming pool. New pools or existing pools (full of water), all shapes & sizes, and any construction type.

9.2. DISADVANTAGES
- It requires a very large amount of data in order to perform better than other techniques.It is extremely expensive to train due to complex data models.
- Moreover, deep learning requires expensive GPUs and hundreds of machines.These increases cost to the users.
- It can take days for a model to learn the parameters that constitute the model.
- Professional-quality underwater cameras can be a pricey investment

10. CONCLUSION

Swimming is one of the best exercises that helps people to reduce stress in this urban lifestyle. Swimming pools are found larger in number in hotels, and weekend tourist spots, and barely people have them in their houses and backyards. Beginners, especially, often feel it difficult to breathe underwater which causes breathing trouble which in turn causes a drowning accident. Worldwide, drowning produces a higher rate of mortality without causing injury to children. Children under six of their age are found to be suffering the highest drowning mortality rates worldwide. Such kinds of deaths account for the third cause of unplanned death globally, with about 1.2 million cases yearly. To overcome this conflict, a meticulous system is to be implemented along the swimming pools to save human life.

Once we have the working drowning detection model we can feed live video footage of the swimming pool to it so that it can keep detecting continuously for any drowning activities. If drowning is detected it will be highlighted on the system screen as well as alarms will be raised to alert security guards so that they can initiate a rescue.

11. FUTURE SCOPE

- Availability of better datasets, modern methodologies, and technologies with high computational power accompanied by high-quality surveillance cameras, will help to improve the accuracy of drowning detection & even can be used in adverse conditions.
- After the implementation of all these essentials, this system also can be used on sea beaches for drowning detection

Note:

The system is not designed to replace a lifeguard or other human monitor, but to act as an additional tool. It helps the lifeguard to detect the underwater situation that they can't easily observe.

12. APPENDIX
12.1. SOURCE CODE
app.py:

```
from flask import Flask
from flask import flash, request, redirect, render_template,
url_for
from cloudant.client import Cloudant
from cvlib.object_detection import draw_bbox


import time
import os
import cvlib as cv
import cv2
import time
import numpy as np


app = Flask(__name__)
client = Cloudant.iam(
    "c56fd99d-acbb-4081-8351-ed1e1e82ba02-bluemix",
    "RKYFESfjUudW-C8Pm-2WaAn3Q9N4Ud49q2PzIbVV4NdU",
    connect=True,
)
db = client["user_details"]
```

```python
@app.after_request
def add_header(r):
    """
    Add headers to both force latest IE rendering engine or
Chrome Frame,
    and also to cache the rendered page for 10 minutes.
    """
    r.headers["Cache-Control"] = "no-cache, no-store, must-
revalidate"
    r.headers["Pragma"] = "no-cache"
    r.headers["Expires"] = "0"
    r.headers["Cache-Control"] = "public, max-age=0"
    return r


@app.route("/")
def home():
    return render_template("index.html", title="VirtualEye -
Home")


@app.route("/login", methods=["GET", "POST"])
def login():
    if request.method == "POST":
        x = [x for x in request.form.values()]
        data = {"_id": x[0], "psw": x[1]}
        query = {"_id": {"$eq": data["_id"]}}
        docs = db.get_query_result(query)
        if len(docs.all()) == 0:
            db.create_document(data)
            return render_template(
                "login.html", title="VirtualEye - Login",
status="NR")
        else:
            if x[0] == docs[0][0]["_id"] and x[1] ==
docs[0][0]["psw"]:
                return redirect(url_for("prediction"))
            else:
                return render_template(
                    "login.html", title="VirtualEye - Login",
status="Failed")
```

```python
        return render_template("login.html", title="VirtualEye -
Login")


@app.route("/register", methods=["GET", "POST"])
def register():
    if request.method == "POST":
        x = [x for x in request.form.values()]
        data = {"_id": x[1], "name": x[0], "psw": x[2]}
        query = {"_id": {"$eq": data["_id"]}}
        docs = db.get_query_result(query)
        if len(docs.all()) == 0:
            db.create_document(data)
            return render_template(
                "register.html", title="VirtualEye - Register",
status="Success")
        else:
            return render_template(
                "register.html", title="VirtualEye - Register",
status="Failed")
    return render_template("register.html", title="VirtualEye -
Register")


@app.route("/demo", methods=["GET"])
def demo():
    return render_template("base.html", title="VirtualEye -
Demo")


@app.route("/forgotpassword")
def forgotpass():
    return render_template("base.html", title="VirtualEye")


@app.route("/logout")
def logout():
    return render_template("logout.html", title="VirtualEye -
Logged out")


@app.route("/result", methods=["GET", "POST"])
def prediction():
    if request.method == "POST" and do_estimation():
        return render_template(
```

```python
                "prediction.html",
                prediction="Emergency!!! The Person is drowning",
                title="VirtualEye - Prediction",)
    return render_template("prediction.html", title="VirtualEye
- Prediction")


def do_estimation():
    webcam = cv2.VideoCapture("drowning.mp4")
    if not webcam.isOpened():
        flash("Could not open webcam")
        exit()


    t0 = time.time()
    centre0 = np.zeros(2)
    isDrowning = False


    while webcam.isOpened():
        status, frame = webcam.read()
        bbox, label, conf = cv.detect_common_objects(frame)


        if len(bbox) > 0:
            centre = [0, 0]
            centre = [(bbox[0][0] + bbox[0][2]) / 2, (bbox[0][1]
+ bbox[0][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, " center:", centre, "
centre0:", centre0)
            print("Are they drowning: ", isDrowning)
```

```python
            centre0 = centre
        out = draw_bbox(frame, bbox, label, conf)
        cv2.imshow("Real-time object detection", out)


        if isDrowning:
            os.system("mpg123 -q alarm.mp3")
            webcam.release()
            cv2.destroyAllWindows()
            return True


        if cv2.waitKey(1) & 0xFF == ord("q"):
            break


    webcam.release()
    cv2.destroyAllWindows()


if __name__ == "__main__":
    app.run(debug=True)
```

base.html:

```html
<!doctype html>
<html lang="en">

<head>
    <!-- Required meta tags -->
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <!-- Tailwind CSS -->
    <link rel="stylesheet" type="text/css"
href="{{url_for('static', filename='css/tailwind.min.css')}}" />
    <title>{{ title }}</title>
</head>
<body>
    <header>
        <nav class="flex items-center justify-between flex-wrap
bg-gray-700 p-6">
            <div class="flex items-center flex-shrink-0 text-
white mr-6 basis-4/5">
```

```html
                    <span class="font-semibold text-xl tracking-
tight">Virtual Eye</span>
            </div>
            <div class="w-full block flex-grow lg:flex lg:items-
center lg:w-auto">
                <div class="text-sm lg:flex-grow font-semibold">
                        {% block navbar %}
                        <a href="{{url_for('home')}}"
class="block mt-4 lg:inline-block lg:mt-0 text-gray-400
hover:text-white mr-4">
                            Home
                        </a>
                        <a href="{{url_for('register')}}"
                            class="block mt-4 lg:inline-block lg:mt-
0 text-gray-400 hover:text-white mr-4">
                            Register
                        </a>
                        <a href="{{url_for('login')}}" class="block
mt-4 lg:inline-block lg:mt-0 text-gray-400 hover:text-white mr-
4">
                            Login
                        </a>
                        <a href="{{url_for('demo')}}" class="block
mt-4 lg:inline-block lg:mt-0 text-gray-400 hover:text-white mr-
4">
                            Demo
                        </a>
                            {% endblock %}
                </div>
            </div>
        </nav>
    </header>
     {% block main %}
     {% endblock %}
</body>
</html>
```

index.html:

```html
{% extends 'base.html' %}
{% block main %}
<img class="w-full h-64 object-cover" src="{{url_for('static',
filename='img/background.jpg')}}">
```

```
<p class="font-bold text-4xl px-8 pt-8 text-blue-500"
style="text-align: center">About Project</p>
<div class="grid grid-cols-2">
    <div class="flex-col bg-blue-700 shadow-xl rounded px-8 pt-
10 pb-8 m-8 mr-4">
        <p class="m-auto text-xl font-semibold mb-4 text-gray-
200">Problem</p>
        <p class="text-gray-300 text-lg">
            Swimming is one of the best exercises that helps
people to reduce stress in this urban lifestyle.
            Swimming pools are found larger in number in
hotels, and weekend tourist spots and barely people have
            them in their house backyard. Beginners,
especially, often feel it difficult to breathe underwater which
            causes breathing trouble which in turn causes a
drowning accident. Worldwide, drowning produces a higher
            rate of mortality without causing injury to
children. Children under six of their age are found to be
            suffering the highest drowning mortality rates
worldwide. Such kinds of deaths account for the third
            cause of unplanned death globally, with about 1.2
million cases yearly. To overcome this conflict, a
            meticulous system is to be implemented along the
swimming pools to save human life.
        </p>
    </div>
    <div class="flex-col bg-blue-700 shadow-xl rounded px-8 pt-
10 pb-8 m-8 ml-4">
    <p class="m-auto text-xl font-semibold mb-4 text-gray-
200">Solution</p>
    <p class="text-gray-300 text-lg">
    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.
    </p>
```

```
</div>
{% endblock %}
```

login.html:

```
{% extends 'base.html' %}
{% block main %}
{% if status == 'NR' %}
<div class="m-4 p-4 text-sm text-yellow-700 bg-yellow-100
rounded-lg dark:bg-yellow-200 dark:text-yellow-800"
role="alert">
    <span class="font-medium">Oops!</span> Looks like you have
not registered yet.
</div>
{% endif %}
{% if status == 'Failed' %}
<div class="m-4 p-4 text-sm text-red-700 bg-red-100 rounded-lg
dark:bg-red-200 dark:text-red-800" role="error">
    <span class="font-medium">Oops!</span> Looks like you have
entered wrong password.
</div>
{% endif %}
<div class="md:flex md:justify-center mb-6 mt-24">
    <form class="bg-white shadow-xl rounded px-8 pt-10 pb-8 mb-
4" method="post">
        <img class="m-auto w-24 h-24 rounded-full mb-8"
src="{{url_for('static', filename='img/logo.png')}}"
alt="Virutal Eye Logo">
        <div class="mb-4">
            <label class="block text-gray-700 text-sm font-bold
mb-2" for="username">
                Email ID
            </label>
            <input
                class="shadow appearance-none border rounded w-
full py-2 px-3 text-gray-700 leading-tight focus:outline-none
focus:shadow-outline"
                id="username" type="email" name="Email ID"
required>
        </div>
        <div class="mb-6">
            <label class="block text-gray-700 text-sm font-bold
mb-2" for="password">
```

```
            Password
        </label>
        <input
            class="shadow appearance-none border rounded w-
full py-2 px-3 text-gray-700 mb-3 leading-tight focus:outline-
none focus:shadow-outline"
            id="password" type="password" name="password"
required>
    </div>
    <div class="mt-6">
        <input
            class="bg-blue-500 hover:bg-blue-700 text-white
font-bold py-2 px-4 rounded focus:outline-none focus:shadow-
outline"
            type="submit" value="Login">
        <a class="inline-block align-baseline font-bold
text-sm text-blue-500 hover:text-blue-800"
            href="{{url_for('forgotpass')}}">
            Forgot Password?
        </a>
    </div>
    </form>
</div>
{% endblock %}
```

logout.html:

```
{% extends 'base.html' %}
{% block main %}
<div class="flex-col items-center">
    <p class="font-semibold text-2xl m-8" style="text-align:
center">Successfully Logged Out!</p>
    <p class="font-semibold text-l mb-4 text-green-500"
style="text-align: center">Login for more information</p>
    <button
        class="block m-auto bg-blue-500 hover:bg-blue-700 text-
white font-bold py-2 px-4 rounded-full">Login</button>
</div>
{% endblock %}
```

prediction.html:

```
{% extends 'base.html' %}
```

```
{% block navbar %}
<a href="{{url_for('home')}}" class="block mt-4 lg:inline-block
lg:mt-0 text-gray-400 hover:text-white mr-4">
    Home
</a>
<a href="{{url_for('logout')}}" class="block mt-4 lg:inline-
block lg:mt-0 text-gray-400 hover:text-white mr-4">
    Logout
</a>
{% endblock %}
{% block main %}
<p class="font-bold text-4xl px-8 pt-8 text-blue-500"
style="text-align: center">VirtualEye - Life Guard for Swimming
    Pools to Detect Active Drowning</p>
<div class="grid grid-cols-2">
    <div class="flex-col bg-blue-700 shadow-xl rounded px-8 pt-
10 pb-8 m-8 mr-4">
        <p class="m-auto text-xl text-gray-200 font-semibold mb-
4">Problem</p>
        <p class="text-gray-300 text-lg">
            Swimming is one of the best exercises that helps
people to reduce stress in this urban lifestyle.
            Swimming pools are found larger in number in hotels,
and weekend tourist spots and barely people have
            them in their house backyard. Beginners, especially,
often feel it difficult to breathe underwater which
            causes breathing trouble which in turn causes a
drowning accident. Worldwide, drowning produces a higher
            rate of mortality without causing injury to
children. Children under six of their age are found to be
            suffering the highest drowning mortality rates
worldwide. Such kinds of deaths account for the third
            cause of unplanned death globally, with about 1.2
million cases yearly. To overcome this conflict, a
            meticulous system is to be implemented along the
swimming pools to save human life.
        </p>
    </div>
    <div class="flex-col bg-white shadow-xl rounded px-8 pt-10
pb-8 m-8 ml-4">
        <img class="h-64 m-auto" src="{{url_for('static',
filename='img/background.jpg')}}">
```

```
        <p class="text-red-500 text-xl text-bold" style="text-
align: center">{{ prediction }}</p>
        <form method="post" action="{{url_for('prediction')}}">
            <input name="demo" type="submit" value="Click me!
For a demo"
                  class="bg-blue-500 hover:bg-blue-700 text-white
font-bold py-2 px-4 rounded-full block m-auto mt-4 mb-4">
        </form>
    </div>
</div>
{% endblock %}
```

register.html:

```
{% extends 'base.html' %}
{% block main %}
{% if status == 'Success' %}
<div class="m-4 p-4 text-sm text-green-700 bg-green-100 rounded-
lg dark:bg-green-200 dark:text-green-800" role="alert">
    <span class="font-medium">Success!</span> Your registration
was successful.
</div>
{% endif %}
{% if status == 'Failed' %}
<div class="m-4 p-4 text-sm text-yellow-700 bg-yellow-100
rounded-lg dark:bg-yellow-200 dark:text-yellow-800"
role="alert">
    <span class="font-medium">Oops!</span> Looks like you have
already registered.
</div>
{% endif %}
<div class="md:flex md:justify-center mb-6 mt-20 flex-wrap">
    <form class="bg-white shadow-xl rounded px-8 pt-10 pb-8 mb-
4" method="post">
        <img class="w-24 h-24 m-auto rounded-full mb-8"
src="{{url_for('static', filename='img/logo.png')}}"
alt="Virutal Eye Logo">
        <div class="mb-4">
            <label class="block text-gray-700 text-sm font-bold
mb-2" for="name">
                Name
            </label>
            <input
```

```
                class="shadow appearance-none border rounded w-
full py-2 px-3 text-gray-700 leading-tight focus:outline-none
focus:shadow-outline"
                id="name" type="text" name="Name">
        </div>
        <div class="mb-4">
            <label class="block text-gray-700 text-sm font-bold
mb-2" for="username">
                Email ID
            </label>
            <input
                class="shadow appearance-none border rounded w-
full py-2 px-3 text-gray-700 leading-tight focus:outline-none
focus:shadow-outline"
                id="username" type="email" name="Email ID">
        </div>
        <div class="mb-6">
            <label class="block text-gray-700 text-sm font-bold
mb-2" for="password">Password</label>
            <input
                class="shadow appearance-none border rounded w-
full py-2 px-3 text-gray-700 mb-3 leading-tight focus:outline-
none focus:shadow-outline"
                id="password" type="password" name="Password">
        </div>
        <div class="mt-6">
            <input class="bg-blue-500 hover:bg-blue-700 text-
white font-bold py-2 px-4 rounded focus:outline-none
focus:shadow-outline"  type="submit" value="Register">
            <a class="inline-block align-baseline font-bold
text-sm text-blue-500 hover:text-blue-800"
                href="{{url_for('login')}}">Have login?</a>
        </div>
    </form>
</div>
{% endblock %}
```

12.2 LINKS

GitHub Link: https://github.com/IBM-EPBL/IBM-Project-665-1658313554
Demo Video Link: https://drive.google.com/file/d/13WBMIiTmuEfa99-dEySZv7oZOzk2swhC/view?usp=share_link