K. K. Wagh Institute Of Engineering Education And Research

REAL TIME OBJECT TRACKING USING ML

Under the guidance of
Prof. P. V. Pandit

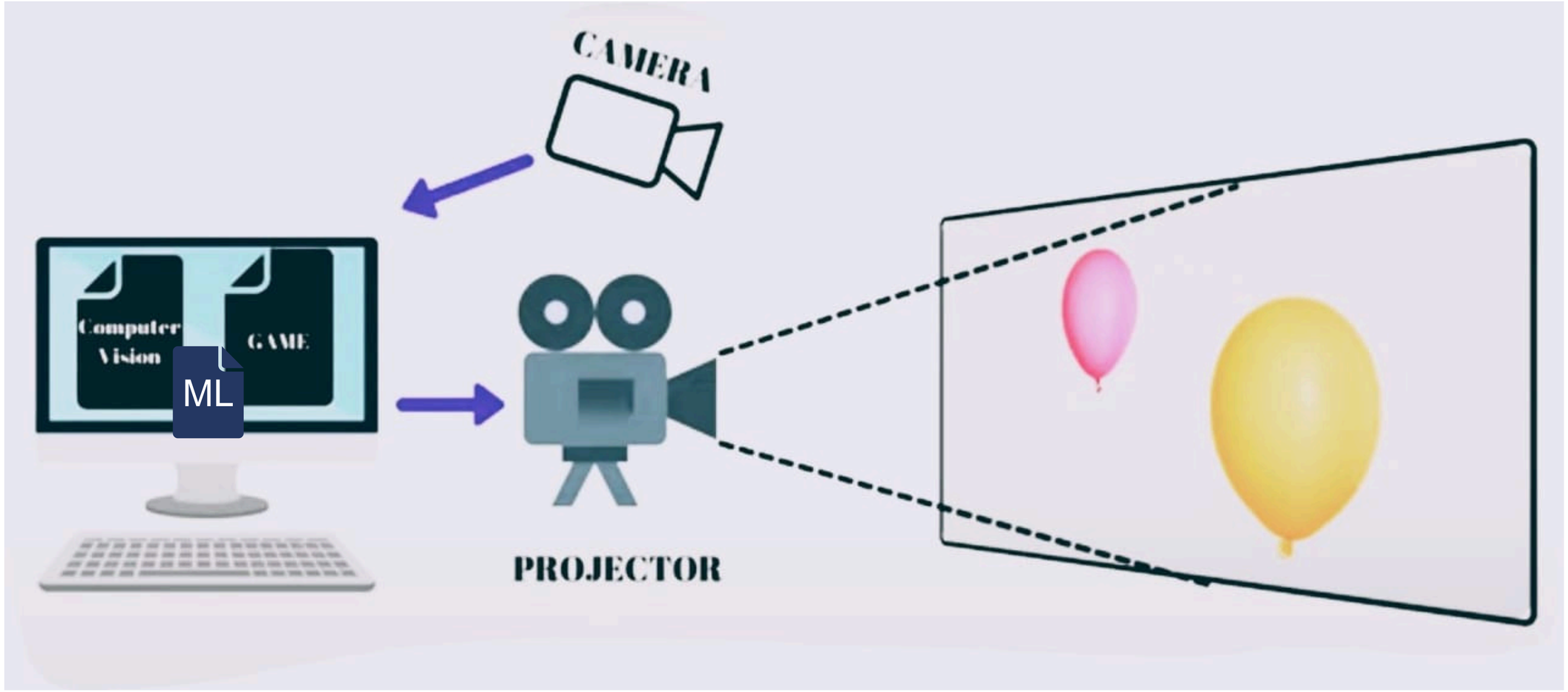
Presented By: Group No. 18

10 Shruti Bhabad
13 Kshitij Bhalerao
48 Prerna Nagare
58 Divya Shinde

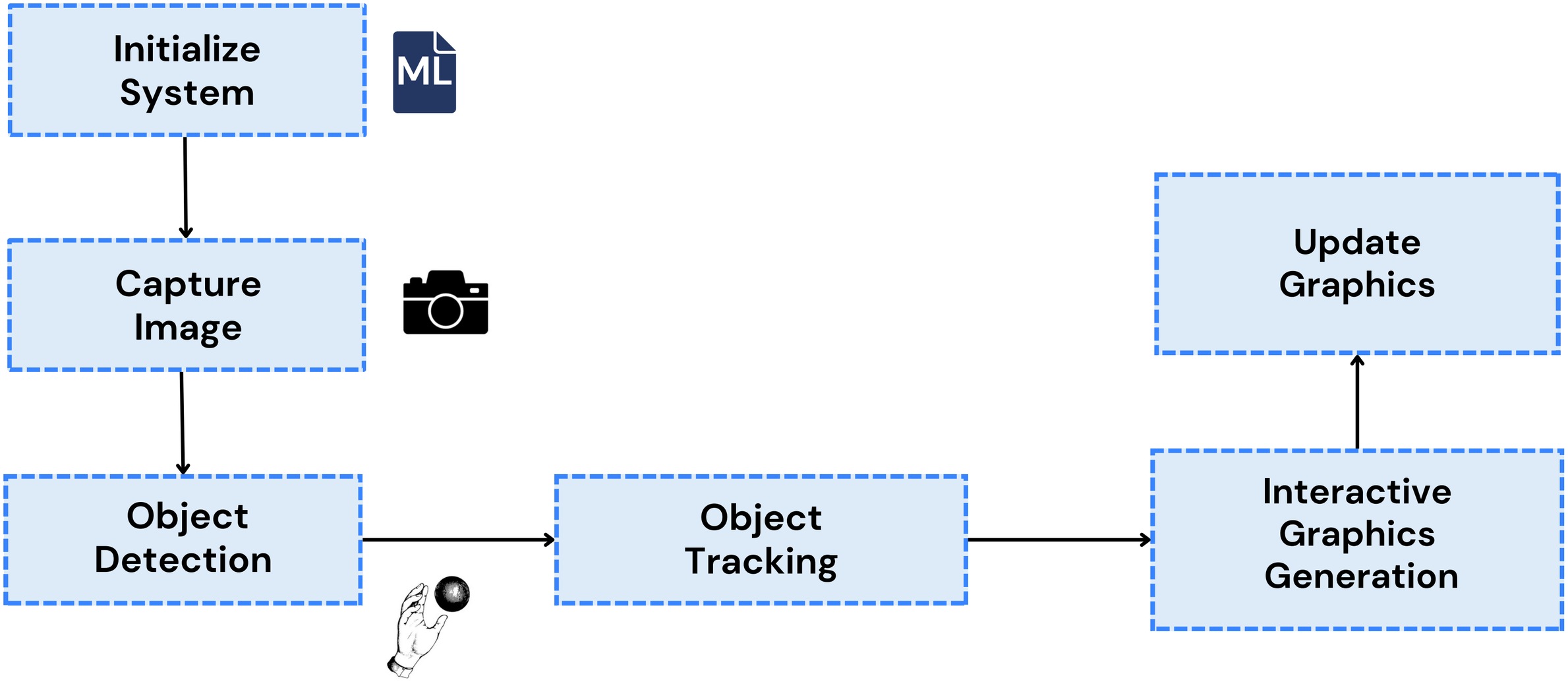
Agenda

1. Abstract
2. Revised Final Design
3. Tools and Techniques Used
4. Partial Implementation
5. Partial Result

Experimental Setup



Revised final design



Tools and Techniques

Tools :

Hardware Tool:

- Camera
- Projection Device

Software Tool & Library:

- Operating System
- Programming Languages:
  - Python
- Machine Learning Framework:
  - YOLO
- Python Game library
  - PyGame

Techniques :

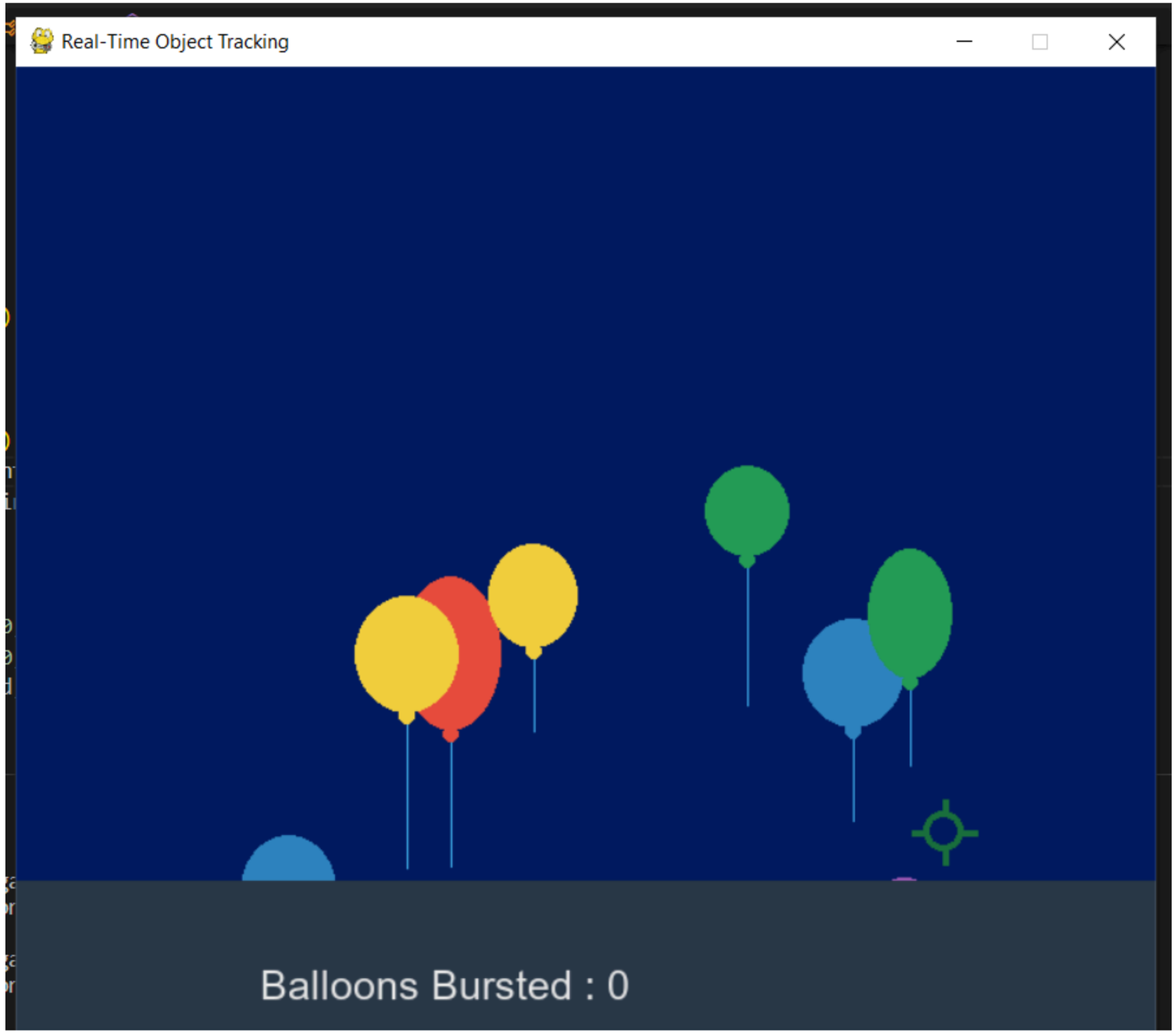
- Object Detection & Tracking
- Image Classification

Partial Implementation

```python
import pygame
import sys
import random
from math import *
pygame.init()

width = 700
height = 600

display = pygame.display.set_mode((width, height))
pygame.display.set_caption("Real-Time Object Tracking")
clock = pygame.time.Clock()

margin = 100
lowerBound = 100
score = 0
white = (230, 230, 230)
lightBlue = (4, 27, 96)
red = (231, 76, 60)
lightGreen = (25, 111, 61)
darkGray = (40, 55, 71)
darkBlue = (64, 178, 239)
green = (35, 155, 86)
yellow = (244, 208, 63)
blue = (46, 134, 193)
purple = (155, 89, 182)
orange = (243, 156, 18)

font = pygame.font.SysFont("Arial", 25)

class Balloon:
    def __init__(self, speed):
        self.a = random.randint(50, 70)
        self.b = self.a + random.randint(0, 40)
        self.x = random.randrange(margin, width - self.a - margin)
        self.y = height - lowerBound
        self.angle = 90
        self.speed = -speed
        self.proPool= [-1, -1, -1, 0, 0, 0, 0, 1, 1, 1]
        self.length = random.randint(50, 100)
        self.color = random.choice([red, green, purple, orange, yellow, blue])

    def move(self):
        direct = random.choice(self.proPool)
```

```
    def move(self):
        direct = random.choice(self.proPool)

        if direct == -1:
            self.angle += -10
        elif direct == 0:
            self.angle += 0
        else:
            self.angle += 10

        self.y += self.speed*sin(radians(self.angle))
        self.x += self.speed*cos(radians(self.angle))

        if (self.x + self.a > width) or (self.x < 0):
            if self.y > height/5:
                self.x -= self.speed*cos(radians(self.angle))
            else:
                self.reset()
        if self.y + self.b < 0 or self.y > height + 30:
            self.reset()

    def show(self):
        pygame.draw.line(display, darkBlue, (self.x + self.a/2, self.y + self.b), (self.x + self.a/2, self.y + self.b + self.length))
        pygame.draw.ellipse(display, self.color, (self.x, self.y, self.a, self.b))
        pygame.draw.ellipse(display, self.color, (self.x + self.a/2 - 5, self.y + self.b - 3, 10, 10))

    def burst(self):
        global score
        pos = pygame.mouse.get_pos()

        if isonBalloon(self.x, self.y, self.a, self.b, pos):
            score += 1
            self.reset()

    def reset(self):
        self.a = random.randint(30, 40)
        self.b = self.a + random.randint(0, 10)
        self.x = random.randrange(margin, width - self.a - margin)
        self.y = height - lowerBound
        self.angle = 90
        self.speed -= 0.002
        self.proPool = [-1, -1, -1, 0, 0, 0, 0, 1, 1, 1]
        self.length = random.randint(50, 100)
```

Partial Result



Reference

[1] William T. Freeman, David B. Anderson, Paul A. Beardsley, Chris N. Dodge, Michal Roth, Craig D. Weissman, and William S. Yerazunis Computer Vision for Interactive Computer Graphics Article in IEEE Computer Graphics and Applications DOI: 10.1109/38.674971

[2] VirtualTable: a projection augmented reality game, A. Dal Corso M. Olsen K. H. Steenstrup J. Wilm S. Jensen R. Paulsen E. Eiriksson,J. Nielsen J. R. Frisvad G. Einarsson H. M. Kjer.Department of Applied Mathematics and Computer Science, Technical University of Denmark.

[3] Markus Löchtefeld,Johannes Schöning, Michael Rohs,Antonio Krüger, LittleProjectedPlanet: An Augmented Reality Game for Camera Projector Phones .

[4] A Dice Game in Third-Person Augmented Reality, Richard Colvin, Ted Hung, David Jimison, Benjamin Johnson, Eben Myers, Tina Blaine Entertainment Technology Center Carnegie Mellon University 700 Technology Drive Pittsburgh, PA I521 9 USA

THANK YOU