

ДОДАТОК А

Лістинг програми для ідентифікації вибухонебезпечних предметів

```
import cv2
from ultralytics import YOLO
import random

def draw_bounding_boxes_without_id(frame, results):
    boxes = results[0].boxes.xyxy.cpu().numpy().astype(int)
    classes = results[0].boxes.cls.cpu().numpy().astype(int)

    for box, cls in zip(boxes, classes):
        if cls != 0:
            random.seed(int(cls)+8)
            color = (random.randint(0, 255), random.randint(0, 255), random.randint(0,
255))

            cv2.rectangle(frame, (box[0], box[1]), (box[2], box[3]), color, 2)
            cv2.putText(
                frame,
                f"{model.model.names[cls]}",
                (box[0], box[1]),
                cv2.FONT_HERSHEY_SIMPLEX,
                0.6,
                (50, 255, 50),
                2,
            )
    return frame
```

```

def process_video_with_tracking(model, input_video_path,
show_video=True, save_video=False, output_video_path="output_video.mp4"):
    cap = cv2.VideoCapture(input_video_path)

    if not cap.isOpened():
        raise Exception("Error: Could not open video file.")

    fps = int(cap.get(cv2.CAP_PROP_FPS))
    frame_width = int(cap.get(cv2.CAP_PROP_FRAME_WIDTH))
    frame_height = int(cap.get(cv2.CAP_PROP_FRAME_HEIGHT))

    if save_video:
        fourcc = cv2.VideoWriter_fourcc(*'mp4v')
        out = cv2.VideoWriter(output_video_path, fourcc, fps, (frame_width,
frame_height))

    while True:
        ret, frame = cap.read()
        if not ret:
            break

        results = model.track(frame, iou=0.4, conf=0.5, persist=True, imgsz=608,
verbose=False, tracker="bytetrack.yaml", classes=0)

        results_detect = model_detect.predict(frame, iou=0.4, conf=0.5, imgsz=608,
verbose=False)

        if results[0].boxes.id != None:
            boxes = results[0].boxes.xyxy.cpu().numpy().astype(int)
            ids = results[0].boxes.id.cpu().numpy().astype(int)

```

```
for box, id in zip(boxes, ids):

    random.seed(int(id))

    color = (random.randint(0, 255), random.randint(0, 255), random.randint(0,
255))

    cv2.rectangle(frame, (box[0], box[1]), (box[2], box[3]), color, 2)

    cv2.putText(
        frame,
        f"Id {id}",
        (box[0], box[1]),
        cv2.FONT_HERSHEY_SIMPLEX,
        0.70,
        (0, 255, 255),
        2,
    )

    if results_detect[0].boxes != None:
        draw_bounding_boxes_without_id(frame, results_detect)
        if save_video:
            out.write(frame)

    if show_video:
        frame = cv2.resize(frame, (0, 0), fx=0.75, fy=0.75)
        cv2.imshow("frame", frame)

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

```
cap.release()
if save_video:
out.release()

cv2.destroyAllWindows()
return results_detect, results

model =
YOLO('D:\\Study4\\Git\\Dyplom\\DyplomOsn\\runs\\detect\\train7\\weights\\best.
pt')
model_detect =
YOLO('D:\\Study4\\Git\\Dyplom\\DyplomOsn\\runs\\detect\\train7\\weights\\best.
pt')
model.fuse()
model_detect.fuse()
results_detect, results = process_video_with_tracking(model, "test.mp4",
show_video=True, save_video=False, output_video_path="output_video.mp4")
```

ДОДАТОК Б

Демонстраційний матеріал у вигляді презентацій

