40)Write a Python function to extract the text from videos.

```
import cv2


def detect_text_regions_from_video(video_path):
    # Open the video file
    cap = cv2.VideoCapture(video_path)

    # Check if the video file can be opened
    if not cap.isOpened():
        print("Error: Could not open video file. Please check the path.")
        return

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

        # Convert the frame to grayscale
        gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)

        # Apply edge detection
        edges = cv2.Canny(gray, 100, 200)

        # Find contours in the edge-detected image
        contours, _ = cv2.findContours(edges, cv2.RETR_EXTERNAL, cv2.CHAIN_APPROX_SIMPLE)

        # Draw bounding boxes around detected contours
        for contour in contours:
            x, y, w, h = cv2.boundingRect(contour)
            # Filter contours based on size to exclude noise
            if w > 50 and h > 20:  # Adjust thresholds as needed
```

```
            cv2.rectangle(frame, (x, y), (x + w, y + h), (0, 255, 0), 2)
            # Display placeholder text above the bounding box
            cv2.putText(frame, "Detected Text", (x, y - 10), cv2.FONT_HERSHEY_SIMPLEX, 0.5, (0, 255,
0), 1)

        # Display the frame with detected regions
        cv2.imshow("Text Region Detection", frame)

        # Exit if 'q' is pressed
        if cv2.waitKey(1) & 0xFF == ord('q'):
            print("Exiting video...")
            break

    # Release the video capture object and close any OpenCV windows
    cap.release()
    cv2.destroyAllWindows()

# ✅ Correct main block
if __name__ == "__main__":
    # Replace with the path to your video file
    video_path = r"C:\Users\harik\Downloads\CV LAB\human.mp4"
    detect_text_regions_from_video(video_path)
```

OUTPUT:


Detected Text
Detected Text
Detected Text
Detected Text
Detected Text
Detected Text