```
import cv2
import mediapipe as mp

# Initialize MediaPipe Pose
mp_drawing = mp.solutions.drawing_utils
mp_pose = mp.solutions.pose

# Path to input video file
input_video = (r"C:\Users\Hanshu\Desktop\photos\Camera\video_20211120_111210.mp4")

# Initialize video capture
cap = cv2.VideoCapture(input_video)

# Get the input video size
width = int(cap.get(cv2.CAP_PROP_FRAME_WIDTH))
height = int(cap.get(cv2.CAP_PROP_FRAME_HEIGHT))

# Initialize BlazePose
with mp_pose.Pose(
    min_detection_confidence=0.5,
    min_tracking_confidence=0.5) as pose:

    # Create a resizable window for display
    cv2.namedWindow('Real-time Body Pose Tracking with BlazePose', cv2.WINDOW_NORMAL)
    cv2.resizeWindow('Real-time Body Pose Tracking with BlazePose', width, height)

    while cap.isOpened():
        # Read frame from video capture
        success, frame = cap.read()
        if not success:
            break
        # Convert the BGR image to RGB
        image = cv2.cvtColor(frame, cv2.COLOR_BGR2RGB)

        # Process the image with BlazePose
        results = pose.process(image)

        if results.pose_landmarks:
            # Render the landmarks on the frame
            mp_drawing.draw_landmarks(
                frame, results.pose_landmarks, mp_pose.POSE_CONNECTIONS)

        # Show the frame
        cv2.imshow('Real-time Body Pose Tracking with BlazePose', frame)

        # Break the loop on 'q' key press
        if cv2.waitKey(5) & 0xFF == ord('q'):
            break

# Release the video capture and close all windows
cap.release()
cv2.destroyAllWindows()
```