```
            # Draw a bounding box around the person and label it as person
detected
            x, y, w, h = cv2.boundingRect(cnt)
            cv2.rectangle(frame, (x, y), (x+w, y+h), (0, 0, 255), 2)
            cv2.putText(frame, 'Person Detected', (x, y-10),
                        cv2.FONT_HERSHEY_SIMPLEX, 0.3, (0, 255, 0), 1,
cv2.LINE_AA)

    # Stack both frames and show the image
    fgmask_3 = cv2.cvtColor(fgmask, cv2.COLOR_GRAY2BGR)
    stacked = np.hstack((fgmask_3, frame))
    cv2.imshow('Combined', cv2.resize(stacked, None, fx=0.65, fy=0.65))

    k = cv2.waitKey(40) & 0xff
    if k == ord('q'):
        break

cap.release()
cv2.destroyAllWindows()
```

multiplecamfeeds.py

```
import cv2
import numpy as np
# initlize video capture object
# capture video from webcam
cap1 = cv2.VideoCapture(0)

# capture video from file
cap2 = cv2.VideoCapture('https://10.143.38.102:8080/video')
cap3 = cv2.VideoCapture('https://192.168.137.66:8080/video')
# cap2 = cv2.VideoCapture(0)
# cap3 = cv2.VideoCapture(0)

# you can set custom kernel size if you want
kernel = None

# initilize background subtractor object
# foog = cv2.createBackgroundSubtractorMOG2(
#       detectShadows=True, varThreshold=50, history=500)
foog = cv2.createBackgroundSubtractorMOG2(
    detectShadows=True, varThreshold=50, history=350)
```