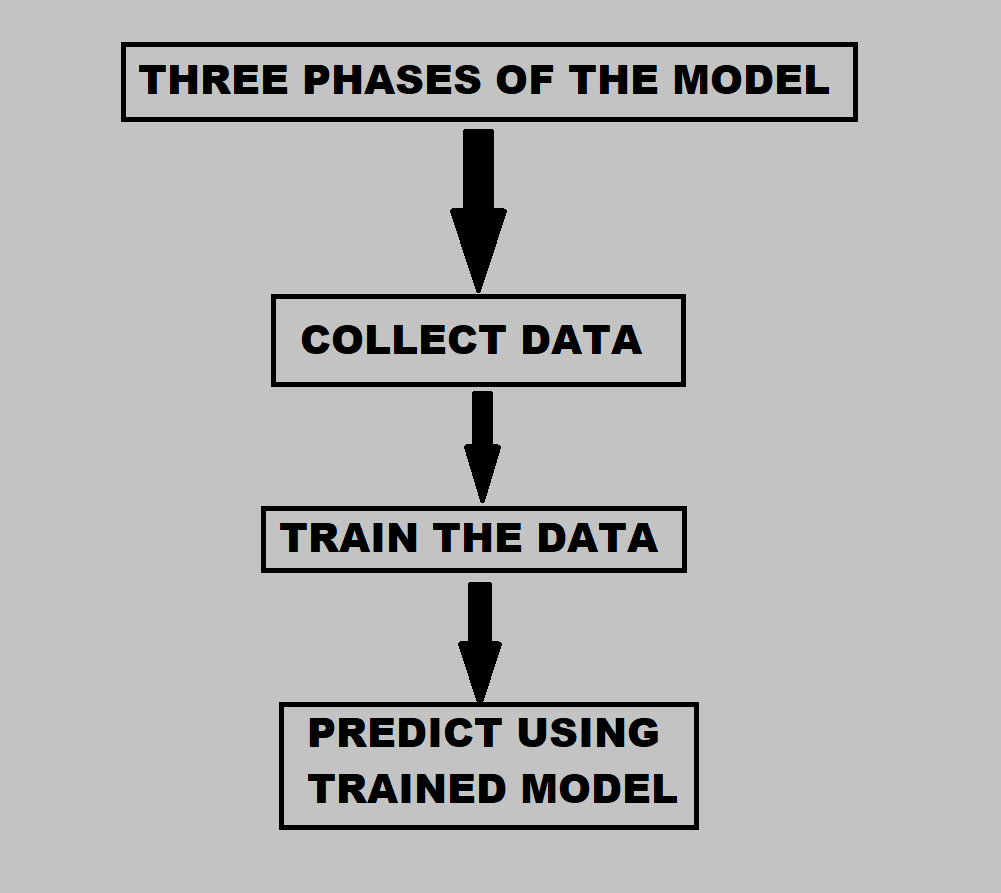
**PHASES OF SYSTEM**

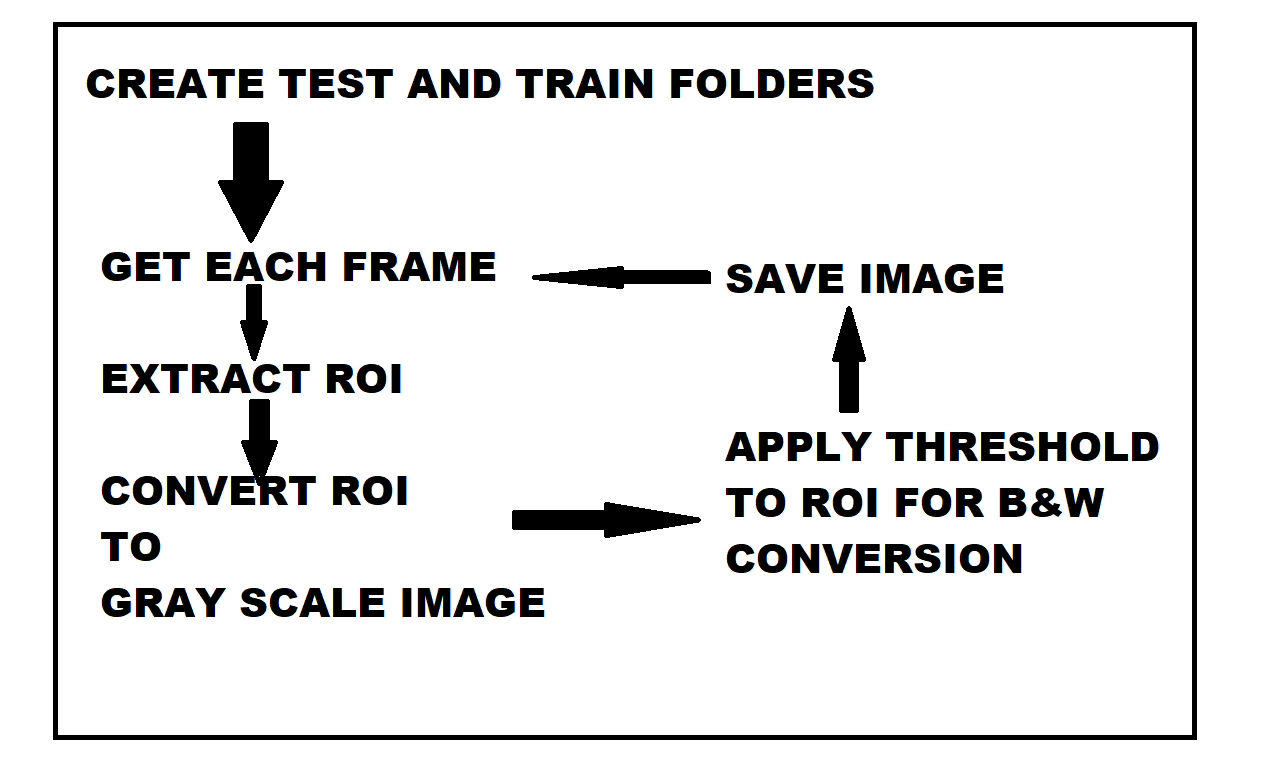
****

**COLLECT DATA**

Libraries used - cv2 & os

**STEPS**

* create folders using os library
* draw a rectangle for extracting region of interest (roi) from each frame
* convert roi to grayscale image
* Feed the new roi to threshold function
* The threshold function gives entire black and white image using thresh\_binary method
* Write the image to the train and test folders



**CREATE DATA CODE SNIPPET**

1. to join folders

os.path.join()

2. creates folder if path not exist

if os.path.exists(path):

os.makedir(path)

3. cap=cv2.VideoCapture(0)

4. while True:

\_,frame=cap.read()

frame=cv2.flip(frame,1)

#we flip frame so that the blue box

#does not fall in front of our face

# 1:y-axis, 0:xaxis, -1:both

cv2.rectangle((x1,y1),(x2,y2),(255,0,0),1)

cv2.rectangle(src,dest,bgr,width)

#draw a rectangle in the frame to capture the gesture

extract roi

resize roi

cv2.imshow("window name",frame)

roi = cv2.cvtColor(roi, cv2.COLOR\_BGR2GRAY)

\_,roi=cv2.cvtColor(roi, 120,255, cv2.THRESH\_BINARY)

0-black 255-white

B\_120\_W

source image grayscale image

threshold value

maxVal

thresh-binary - convert into 0 and black

cv2.imwrite(path of train/test,roi)

PRESS ESCAPE KEY TO EXIT WINDOW

interrupt = cv2.waitKey(10)

if interrupt & 0xFF == 27:

break

5. cap.release()

cv2.destryAllWindows()