background.py

#1.get a background image

#2.select a colour; we are choosing red

import cv2

cap=cv2.VideoCapture(0) #capture photo from my camera –source=0 for camera

while cap.isOpened():# while my capture is opened

ret, back = cap.read() #true if able to read the image else false

if ret:

#back is image what the camera is reading

#ret is telling whether image reading was successful or not(i.e camera is working fine or not)

cv2.imshow("image",back)

#show the i mage is capturing

if cv2.waitKey(5) == ord('q'):#press q to click the image;

#waitKey waits for a particular amount of time,after that when you press anything it is compared with rhs;5ms is the refresh rate; after every 5 sec image is captured

#save the image

cv2.imwrite('image.jpg', back)

#writing the captured image(back) into a file:image.jpg in the same directory

Break

#click the image and break out of loop

cap.release() #release all the resources

cv2.destroyAllWindows() #destroy all the windows

invisible\_cloak.py

import cv2

import numpy as np

cap=cv2.VideoCapture(0)

back=cv2.imread('./image.jpg')#read the background image

#WHY HSV?

#hsv= h:huerepresents color, s:saturation represents in what amount color is mixed with white #color, v:value represent in what amount color is mixed with black color

#hsv model describes color how the human eye perceives the color; light and its intensity matters #here

#hsv is mixture of color and how much light and intensity is projected on object

#rgb model describes color as combination of many other primary colors

#rgb is just the mixture of colors

#so we use hsv instead of rgb for object detection

while cap.isOpened():

#take each frame

ret,frame=cap.read()

if ret:

#how do we convert rgb to hsv

hsv=cv2.cvtColor(frame,cv2.COLOR\_BGR2HSV)

#cv2.imshow("hsv",hsv) #we get hsv image here

#how to get hsv value

#lower:hue-10, 100, 100 ; higher:hue+10, 255,255

red=np.uint8([[[0,0,255]]]) #B:blue, g:green, r:red value

hsv\_red=cv2.cvtColor(red,cv2.COLOR\_BGR2HSV)

#print(hsv\_red)

#threshold the hsv value to get only red colors

#l\_red=np.array([0,100,100])

#u\_red=np.array([10,255,255])

l\_red=np.array([161,155,84]) #gives more better result

u\_red=np.array([179,255,255])

mask=cv2.inRange(hsv,l\_red,u\_red)

#our task is to disappear all color that fall in given range

cv2.imshow('mask',mask) #just highlights the red color; ignore others

#part1:all things are red

part1 = cv2.bitwise\_and(back,back,mask=mask)

#doing and : background image and mask created

#cv2.imshow("part1",part1) #replace red color with background image

mask=cv2.bitwise\_not(mask)

#creating mask for not the red color; i.e opposite of previous mask

#now I want face to be shown; we need to display all the things which is not red

#so create mask which is not the previous mask

#part2: all things not red

part2=cv2.bitwise\_and(frame,frame,mask=mask)

#cv2.imshow("mask",part2)

cv2.imshow("cloak",part1+part2)#I want all things red and not red

#https://docs.opencv.org/master/d9/d61/tutorial\_py\_morphological\_ops.html

#try morphology for better view

if cv2.waitKey(5)==ord('q'):

#save the image

#cv2.imwrite('frame.jpg', frame)

break

cap.release()

cap.destroyAllWindows()