CODE

import cv2

import numpy as np

import time

w=0.95

frameWidth = 1920

frameHeight = 1080

cap = cv2.VideoCapture("hazy\_video.mp4")

out = cv2.VideoWriter('dehazed.mp4',cv2.VideoWriter\_fourcc(\*'MP4V'), 20.0, (1920,1080))

cap.set(3, frameWidth)

cap.set(4, frameHeight)

cap.set(10, 150)

frames=0

success=True

while success==True:

start = time.time()

frames=frames+1

success, img=cap.read()

dark=cv2.cvtColor(img,cv2.COLOR\_BGR2GRAY)

dark = cv2.cvtColor(img, cv2.COLOR\_BGR2GRAY)

b, g, r = cv2.split(img)

dark = np.minimum(r, g) # find dark channel

dark = np.minimum(dark, b) # -----------------

A = np.amax(dark) # atmospheric light= max of dark channel(used while changing sand v values in hsv)

kernel = np.ones((5, 5), np.uint8)

dark = cv2.erode(dark, kernel) # use erode so output does not look pixelated

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

hsv[:, :, 1] = hsv[:, :, 1] // (1 - dark[:, :] / A \* 0.8) +10 # increase saturation by more if pixel more hazy

hsv[:, :, 2] = hsv[:, :, 2] // (1 + dark[:, :] / A \* 0.5) - 20 # increase value by more if pixel smore hazy

"""filter = hsv[:, :, 2] < 50 # increase value if its too low

hsv[filter, 2] = 50"""

filter = hsv[:, :, 1] > 100 # decrease saturation if its too high

hsv[filter, 1] = 50 + hsv[filter, 1] // 2

img = cv2.cvtColor(hsv, cv2.COLOR\_HSV2BGR)

end=time.time()

fps=1/(end-start)

cv2.putText(img,str(fps),(100,100),cv2.FONT\_HERSHEY\_COMPLEX,2,(0,0,255),2)

print(fps)

cv2.imshow("video",img)

out.write(img)

if cv2.waitKey(1) & 0xFF == ord('q'):

break

print(frames)

cap.release()

cv2.waitKey(0)

Dehazing part is same as b4