

```

In [ ]: import numpy as np
import cv2 as cv

cap = cv.VideoCapture('https://www.bogotobogo.com/python/OpenCV_Python/images/mean_

feature_params = dict(maxCorners=100,
                      qualityLevel=0.3,
                      minDistance=7,
                      blockSize=7)

lk_params = dict(winSize=(15, 15),
                maxLevel=2,
                criteria=(cv.TERM_CRITERIA_EPS | cv.TERM_CRITERIA_COUNT, 10, 0.03))
color = np.random.randint(0, 255, (100, 3))
ret, old_frame = cap.read()
old_gray = cv.cvtColor(old_frame, cv.COLOR_BGR2GRAY)
p0 = cv.goodFeaturesToTrack(old_gray, mask=None, **feature_params)
mask = np.zeros_like(old_frame)

while True:
    ret, frame = cap.read()
    if not ret:
        print('No frames grabbed!')
        break
    frame_gray = cv.cvtColor(frame, cv.COLOR_BGR2GRAY)
    p1, st, err = cv.calcOpticalFlowPyrLK(old_gray, frame_gray, p0, None, **lk_params)
    if p1 is not None:
        good_new = p1[st == 1]
        good_old = p0[st == 1]
        for i, (new, old) in enumerate(zip(good_new, good_old)):
            a, b = new.ravel()
            c, d = old.ravel()
            mask = cv.line(mask, (int(a), int(b)), (int(c), int(d)), color[i].tolist(), 2)
            frame = cv.circle(frame, (int(a), int(b)), 5, color[i].tolist(), -1)
        img = cv.add(frame, mask)
        cv.imshow('frame', img)
        k = cv.waitKey(30) & 0xff
        if k == 27:
            break
        old_gray = frame_gray.copy()
        p0 = good_new.reshape(-1, 1, 2)

cv.destroyAllWindows()
cap.release()

```



```
In [ ]: import numpy as np
import cv2 as cv
cap = cv.VideoCapture('https://www.bogotobogo.com/python/OpenCV_Python/images/mean_
ret, frame1 = cap.read()
prvs = cv.cvtColor(frame1, cv.COLOR_BGR2GRAY)
hsv = np.zeros_like(frame1)
hsv[..., 1] = 255
while(1):
    ret, frame2 = cap.read()
    if not ret:
        print('No frames grabbed!')
        break
    else :
        print('break frame')

    next = cv.cvtColor(frame2, cv.COLOR_BGR2GRAY)
    flow = cv.calcOpticalFlowFarneback(prvs, next, None, 0.5, 3, 15, 3, 5, 1.2, 0)
    mag, ang = cv.cartToPolar(flow[..., 0], flow[..., 1])
    hsv[..., 0] = ang*180/np.pi/2
    hsv[..., 2] = cv.normalize(mag, None, 0, 255, cv.NORM_MINMAX)
    bgr = cv.cvtColor(hsv, cv.COLOR_HSV2BGR)
    cv.imshow('frame2', bgr)
    k = cv.waitKey(30) & 0xff
    if k == 27:
        break
    elif k == ord('s'):
        cv.imwrite('opticalfb.png', frame2)
        cv.imwrite('opticalhsv.png', bgr)
    prvs = next

cv.destroyAllWindows()
```



In []: