## **เริ่มต้นใช้งานวิดีโอ**

เป้าหมาย

## เรียนรู้การอ่านวิดีโอแสดงวิดีโอและบันทึกวิดีโอ

## เรียนรู้การถ่ายภาพจากกล้องและแสดงภาพ

## คุณจะได้เรียนรู้ฟังก์ชันเหล่านี้: **cv2.VideoCapture ()** , **cv2.VideoWriter ()**

จับภาพวิดีโอจากกล้อง

## บ่อยครั้งที่เราต้องจับภาพสตรีมสดพร้อมกับกล้อง OpenCV มีอินเทอร์เฟซที่ง่ายมากในการนี้ ลองจับภาพวิดีโอจากกล้อง (ฉันใช้เว็บแคมในตัวของแล็ปท็อป) แปลงเป็นวิดีโอสีเทาและแสดงภาพ เพียงแค่เริ่มต้นใช้งานง่ายๆ

## ในการจับภาพวิดีโอ คุณต้องสร้างออบเจกต์ **VideoCapture** อาร์กิวเมนต์ของมันอาจเป็นได้ทั้งดัชนีอุปกรณ์หรือชื่อของไฟล์วิดีโอ ดัชนีอุปกรณ์เป็นเพียงตัวเลขที่ระบุว่ากล้องใด ปกติแล้วกล้องตัวเดียวจะเชื่อมต่อกัน (เช่นในกรณีของฉัน) ดังนั้นฉันก็ผ่าน **0** (หรือ **-1**) คุณสามารถเลือกกล้องตัวที่สองได้โดยผ่าน **1** และอื่น ๆ หลังจากนั้นคุณสามารถจับภาพแบบทีละเฟรม แต่ในตอนท้ายอย่าลืมปล่อยการจับภาพ

**import** **numpy** **as** **np**

**import** **cv2**

cap = cv2.VideoCapture(0)

**while**(True):

*# Capture frame-by-frame*

ret, frame = cap.read()

*# Our operations on the frame come here*

gray = cv2.cvtColor(frame, cv2.COLOR\_BGR2GRAY)

*# Display the resulting frame*

cv2.imshow('frame',gray)

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

**break**

*# When everything done, release the capture*

cap.release()

cv2.destroyAllWindows()

**cap.read ()** ส่งกลับค่า **bool (True / False)**หากเฟรมอ่านอย่างถูกต้องก็จะเป็น True คุณสามารถตรวจสอบจุดสิ้นสุดของวิดีโอได้โดยการตรวจสอบค่าที่ส่งคืนนี้

บางครั้งอาจไม่ได้เริ่มต้นการจับภาพ ในกรณีนี้รหัสนี้แสดงข้อผิดพลาด คุณสามารถตรวจสอบว่าจะเริ่มต้นได้หรือไม่ได้โดยวิธีการ **cap.isOpened ()** ถ้าเป็นจริงตกลง มิฉะนั้นเปิดใช้ **cap.open ()**

นอกจากนี้คุณยังสามารถเข้าถึงคุณลักษณะบางอย่างของวิดีโอนี้ได้โดยใช้เมธอด **cap.get (propId)**โดยที่ **propId** คือตัวเลขตั้งแต่ 0 ถึง 18 แต่ละหมายเลขจะระบุถึงคุณสมบัติของวิดีโอ (หากใช้ได้กับวิดีโอนั้น) และรายละเอียดทั้งหมด จะเห็นได้ที่นี่: Property Identifier.บางส่วนของค่าเหล่านี้สามารถแก้ไขได้โดยใช้ **cap.set(propId, value)** ค่าคือค่าใหม่ที่คุณต้องการ

ตัวอย่างเช่นผมสามารถตรวจสอบความกว้างและความสูงของกรอบโดย **cap.get (3)** และ **cap.get (4)** เป็นค่าเริ่มต้น **640x480** แต่ฉันต้องการปรับเปลี่ยนเป็น **320x240** เพียงแค่ใช้ **ret = cap.set (3,320)** และ**ret = cap.set (4,240)**

กำลังเล่นวิดีโอจากไฟล์

## เหมือนกับการบันทึกจากกล้องเพียงแค่เปลี่ยนดัชนีกล้องกับชื่อไฟล์วิดีโอ นอกจากนี้ในขณะที่แสดงกรอบให้ใช้เวลาที่เหมาะสมสำหรับ **cv2.waitKey ()** ถ้าวิดีโอมีจำนวนน้อยเกินไปวิดีโอจะเร็วมากและถ้าวิดีโอสูงเกินไปวิดีโอจะช้า (ดังนั่นคือวิธีที่คุณสามารถแสดงวิดีโอในแบบช้า) 25 มิลลิวินาทีจะเป็นปกติในกรณีปกติ

**import** **numpy** **as** **np**

**import** **cv2**

cap = cv2.VideoCapture('vtest.avi')

**while**(cap.isOpened()):

ret, frame = cap.read()

gray = cv2.cvtColor(frame, cv2.COLOR\_BGR2GRAY)

cv2.imshow('frame',gray)

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

**break**

cap.release()

cv2.destroyAllWindows()

การบันทึกวิดีโอ

## เราจึงจับภาพวิดีโอประมวลผลทีละเฟรมและเราต้องการบันทึกวิดีโอดังกล่าว สำหรับภาพมันเป็นเรื่องง่ายมากเพียงใช้ **cv2.imwrite ()** นี่เป็นงานที่ต้องใช้อีกเล็กน้อย

ขณะนี้เราสร้างออบเจกต์ **VideoWriter**  เราควรระบุชื่อไฟล์เอาต์พุต **(เช่น: output.avi)** จากนั้นเราควรระบุรหัส **FourCC** (รายละเอียดในย่อหน้าถัดไป) จำนวนเฟรมต่อวินาที **(fps)** และขนาดเฟรมควรผ่าน และสุดท้ายคือ **isColor** flag. ถ้าเป็น **True encoder** คาดว่ากรอบสีจะทำงานร่วมกับกรอบสีเทา

[**FourCC**](http://en.wikipedia.org/wiki/FourCC) เป็น[รหัส](http://en.wikipedia.org/wiki/FourCC) 4 ไบต์ที่ใช้เพื่อระบุตัวแปลงสัญญาณวิดีโอ รายการของรหัสที่มีอยู่สามารถพบได้ใน**fourcc.org**มันขึ้นอยู่กับแพลตฟอร์ม ตัวแปลงสัญญาณต่อไปนี้ทำงานได้ดี

* ใน Fedora: DIVX, XVID, MJPG, X264, WMV1, WMV2 (XVID เป็นที่นิยมมากขึ้นผล MJPG ในวิดีโอขนาดใหญ่ X264 ให้วิดีโอขนาดเล็กมาก)
* ใน Windows: DIVX (เพิ่มเติมเพื่อทดสอบและเพิ่ม)
* ใน OSX: (ฉันไม่มี OSX สามารถกรอกข้อมูลนี้ได้หรือไม่?)

รหัส FourCC ถูกส่งผ่านไปเป็นcv2.VideoWriter\_fourcc ('M', 'J', 'P', 'G')หรือcv2.VideoWriter\_fourcc (\* 'MJPG)สำหรับ MJPG

ด้านล่างการจับภาพโค้ดจากกล้องถ่ายรูปให้พลิกเฟรมทุกเฟรมตามทิศทางแนวตั้งและบันทึกภาพ

**import** **numpy** **as** **np**

**import** **cv2**

cap = cv2.VideoCapture(0)

*# Define the codec and create VideoWriter object*

fourcc = cv2.VideoWriter\_fourcc(\*'XVID')

out = cv2.VideoWriter('output.avi',fourcc, 20.0, (640,480))

**while**(cap.isOpened()):

ret, frame = cap.read()

**if** ret==True:

frame = cv2.flip(frame,0)

*# write the flipped frame*

out.write(frame)

cv2.imshow('frame',frame)

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

**break**

**else**:

**break**

*# Release everything if job is finished*

cap.release()

out.release()

cv2.destroyAllWindows()