Pengenalan Visual Object Tracking dengan OpenCV

Automation Webinar: Visual Object Tracking & Sensor Fusion

Naser Jawas

Institut Teknologi dan Bisnis STIKOM Bali

December 10, 2020

Outline

1. **Visual Object Tracking**
2. **OpenCV**
3. **Implementasi dengan OpenCV dan Python**
   - Memproses Data
   - Metode Visual Object Tracking di OpenCV

Visual Object Tracking

Visual Object Tracking (VOT) adalah salah satu bagian dari ilmu Computer Vision yang bertujuan untuk membuat komputer dapat mengikuti satu atau lebih objek di sebuah video / deret citra.

OpenCV

OpenCV adalah sebuah pustaka open source yang memiliki sekumpulan implementasi metode-metode computer vision. Pustaka ini ditulis dalam bahasa C/C++ dan dapat digunakan dalam bahasa C/C++ dan juga tersedia binding untuk dapat digunakan dengan bahasa Python, Java, hingga javascript. Selain itu, OpenCV juga dapat berjalan di berbagai platform seperti Linux, Windows, dan MacOS.

Implementasi dengan OpenCV dan Python

Untuk dapat menggunakan library OpenCV dan Python, kita perlu menginstall OpenCV dan beserta package opencv-python melalui pip.

Data

Data yang umumnya diproses dapat berupa:

- Deret Citra
- Video
- Camera (contohnya: Raspberry Pi Camera)

Mengakses Data Deret Citra

Sintaks

```
import glob
import cv2 as cv
...
dirname = "./dirname/"
filenames = sorted(glob.glob(dirname + "*.png"))
images = [cv.imread(filename) for filename in filenames]
...
for image in images:
  # processing per frame
```

Mengakses Data Video

Sintaks

```
import glob
import cv2 as cv

videofile = "./video.mp4"
video = cv.VideoCapture(videofile)

while(video.isOpened()):
  ret, image = video.read()
  if not ret:
    break
  # processing per frame
```

Mengakses Data Raspberry Pi Camera

Sintaks

```
from picamera.array import PiRGBArray
from picamera import PiCamera
import cv2 as cv
import time

camera = PiCamera()
camera.resolution = (640, 480)
camera.framerate = 32
rawCapture = PiRGBArray(camera, size=(640, 480))
time.sleep(0.1)

for frame in camera.capture_continuous(rawCapture, format="bgr",
  use_video_port=True):
   image = frame.array
   # processing per frame
```

Metode Visual Object Tracking di OpenCV

Metode Visual Object Tracking yang tersedia di OpenCV hingga saat ini ada 8 yaitu:

- KCF (Kernelised Correlation Filter) [1]

Sintaks

```
tracker = cv.TrackerKCF_create()
```

- MIL (Multiple Instance Learning) [2]

Sintaks

```
tracker = cv.TrackerMIL_create()
```

Metode Visual Object Tracking di OpenCV (Cont'd)

- TLD (Tracking, Learning and Detection) [3]

Sintaks

```
tracker = cv.TrackerTLD_create()
```

- CSRT (Channel and Spatial Reliability) [4]

Sintaks

```
tracker = cv.TrackerCSRT_create()
```

Metode Visual Object Tracking di OpenCV (Cont'd)

- MOSSE (Minimum Output Sum of Squared Error) [5]

Sintaks

```
tracker = cv.TrackerMOSSE_create()
```

- GOTURN (Generic Object Tracking Using Regression Networks) [6]

Sintaks

```
tracker = cv.TrackerGOTURN_create()
```

Metode Visual Object Tracking di OpenCV (Cont'd)

- Boosting [7]

Sintaks

```
tracker = cv.TrackerBoosting_create()
```

- Median Flow [8]

Sintaks

```
tracker = cv.TrackerMedianFlow_create()
```

Tracking Per Frame

Untuk menjalankan operasi tracking, di awal frame kita cukup mendefinisikan objek pertama yang ingin diikuti dengan memberikan lokasi dari objek tersebut.

Sintaks

```
tracker.init(image, initiallocation)
```

Selanjutnya di setiap frame berikutnya kita cukup memanggil method update untuk meminta update lokasi terbaru dari objek tersebut.

Sintaks

```
(success, box) = tracker.update(image)
```

References I


J. F. Henriques, R. Caseiro, P. Martins, and J. Batista. Exploiting the circulant structure of tracking-by-detection with kernels. In proceedings of the European Conference on Computer Vision, 2012.


Boris Babenko, Ming-Hsuan Yang, and Serge Belongie. Visual tracking with online multiple instance learning. In Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on, pages 983–990. IEEE, 2009.


Zdenek Kalal, Krystian Mikolajczyk, and Jiri Matas. Tracking-learning-detection. Pattern Analysis and Machine Intelligence, IEEE Transactions on, 34(7):1409–1422, 2012.


Alan Lukezic, Tom'as Voj'ir, Luka Cehovin Zajc, Jir'i Matas, and Matej Kristan. Discriminative correlation filter tracker with channel and spatial reliability. International Journal of Computer Vision, 2018.


David S. Bolme, J. Ross Beveridge, Bruce A. Draper, and Man Lui Yui. Visual object tracking using adaptive correlation filters. In Conference on Computer Vision and Pattern Recognition (CVPR), 2010.


David Held, Sebastian Thrun, and Silvio Savarese. Learning to track at 100 fps with deep regression networks. In European Conference Computer Vision (ECCV), 2016.

References II

Helmut Grabner, Michael Grabner, and Horst Bischof. Real-time tracking via on-line boosting. In BMVC, volume 1, page 6, 2006.

Zdenek Kalal, Krystian Mikolajczyk, and Jiri Matas. Forward-backward error: Automatic detection of tracking failures. In Pattern Recognition (ICPR), 2010 20th International Conference on, pages 2756–2759. IEEE, 2010.

Terima Kasih