Branch: master ▾ Find file Copy path

appliedai / SelfDrivingCar / **run_dataset.py** / Jump to ▾

sunneysood Add files via upload

0470b4d 4 minutes ago

1 contributor

Raw Blame History

81 lines (64 sloc) 3.03 KB

You're using code navigation to jump to definitions or references. Learn more or give us feedback

```
#pip3 install opencv-python

#https://stackoverflow.com/questions/37383812/tensorflow-module-object-has-no-attribute-placeholder
import tensorflow.compat.v1 as tf
tf.disable_v2_behavior()

import scipy.misc
import model
import cv2
from subprocess import call
import math, os
import numpy as np

sess = tf.InteractiveSession()
saver = tf.train.Saver()
saver.restore(sess, "save/model.ckpt")

img = cv2.imread('steering_wheel_image.jpg',0)
rows,cols = img.shape

smoothed_angle = 0


#read data.txt
xs = []
ys = []
with open("driving_dataset/data.txt") as f:
    for line in f:
        xs.append("driving_dataset/" + line.split()[0])
        #the paper by Nvidia uses the inverse of the turning radius,
        #but steering wheel angle is proportional to the inverse of turning radius
        #so the steering wheel angle in radians is used as the output
        ys.append(float(line.split()[1]) * scipy.pi / 180)

#get number of images
num_images = len(xs)


i = math.ceil(num_images*0.7)
reduced = i + 50
print("Starting frameofvideo:" +str(i))

#https://stackoverflow.com/questions/9041681/opencv-python-rotate-image-by-x-degrees-around-specific-point
def rotateImage(image, angle):
  image_center = tuple(np.array(image.shape[1::-1]) / 2)
  rot_mat = cv2.getRotationMatrix2D(image_center, angle, 1.0)
  result = cv2.warpAffine(image, rot_mat, image.shape[1::-1], flags=cv2.INTER_LINEAR)
  return result


```

```
#while(cv2.waitKey(10) != ord('q')):
while i < num_images:
    full_image = scipy.misc.imread("driving_dataset/" + str(i) + ".jpg", mode="RGB")
    image = scipy.misc.imresize(full_image[-150:], [66, 200]) / 255.0
    degrees = model.y.eval(feed_dict={model.x: [image], model.keep_prob: 1.0})[0][0] * 180.0 / scipy.pi
    #call("clear")
    #print("Predicted Steering angle: " + str(degrees))
    print("Steering angle: " + str(degrees) + " (pred)\t" + str(ys[i]*180/scipy.pi) + "(actual)")
    #cv2.imshow("frame", cv2.cvtColor(full_image, cv2.COLOR_RGB2BGR))
    # Commenting this line as gcp can't display images
    #https://stackoverflow.com/questions/52644763/gcp-aws-instances-not-working-with-opencv-imshow

    #make smooth angle transitions by turning the steering wheel based on the difference of the current angle
    #and the predicted angle
    smoothed_angle += 0.2 * pow(abs((degrees - smoothed_angle)), 2.0 / 3.0) * (degrees - smoothed_angle) / abs(degrees - smoothed_an
    M = cv2.getRotationMatrix2D((cols/2,rows/2),-smoothed_angle,1)
    dst = cv2.warpAffine(img,M,(cols,rows))

    #https://stackoverflow.com/questions/41586429/opencv-saving-images-to-a-particular-folder-of-choice/41587740
    path = 'sample/'
    output_file_name = "str_"+ str(i) + ".jpg"
    cv2.imwrite(os.path.join(path , output_file_name), dst)

    #cv2.imwrite(os.path.join(path , output_file_name), rotateImage(img,degrees))
    #cv2.waitKey(0)
    # Commenting this line as gcp can't display images
    #https://stackoverflow.com/questions/52644763/gcp-aws-instances-not-working-with-opencv-imshow
    #cv2.imshow("steering wheel", dst)
    i += 1

cv2.destroyAllWindows()
```