Wykład Specjalizujący

Zadanie 4

Oznakowanie linii drogowych przez OpenCV

Autor:

Jerzy Dębowski 151266

Obraz

Oryginalny obraz jest wycinkiem z filmu pod adresem:
https://www.youtube.com/watch?v=jwBaGY67olI

Przekonwertowanie obrazu na czarno-biały

Odczytujemy dany obraz dodając na niego nakładkę cv2.IMREAD_GRAYSCALE, a następnie zapisujemy:

```
def convert_to_gray(img_path, new_img_path):
    gray_img = cv2.imread(img_path, cv2.IMREAD_GRAYSCALE)
    cv2.imshow( winname: 'Grayscale', gray_img)
    cv2.imwrite(new_img_path, gray_img)
```

Rozmycie przez filtr Gaussa

Nakładamy na odczytany obraz nakładkę cv2.GaussianBlur z atrybutami kernel_size = (15,15) oraz sigmaX = 0, a następnie zapisujemy:

```
1 usage   jerzy
def gauss(img_path, new_img_path):
    blur_kernel_size = (15, 15)
    gray_img = cv2.imread(img_path)
    gray_blur = cv2.GaussianBlur(gray_img, blur_kernel_size, sigmaX: 0)
    cv2.imshow( winname: 'Grayscale blur', gray_blur)
    cv2.imwrite(new_img_path, gray_blur)
```

Dobór konturów

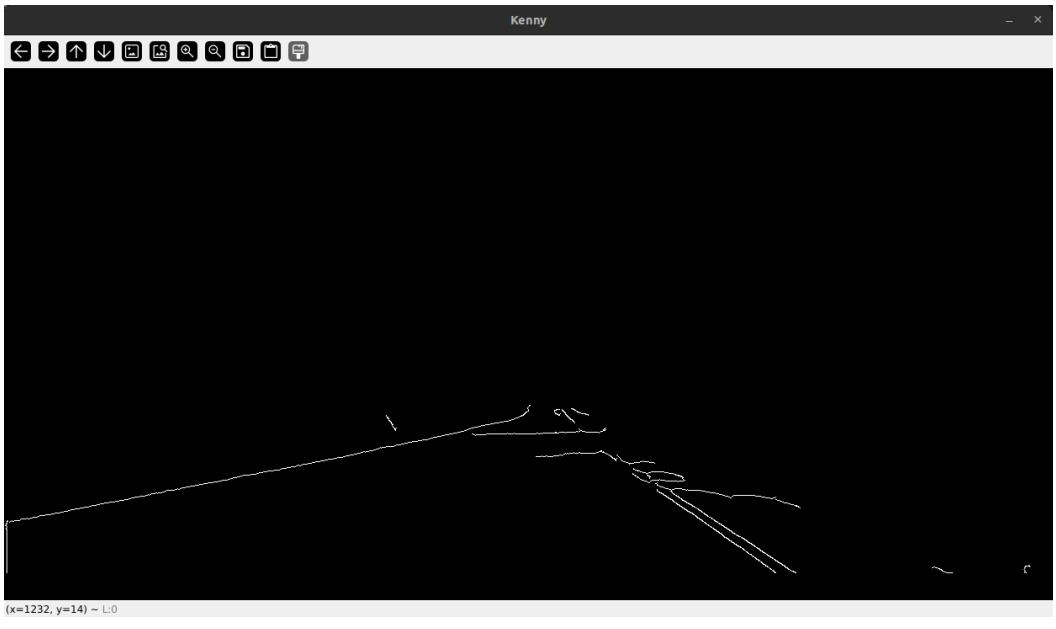
Na zapisany wcześniej obraz dobieramy konturów używając algorytmu Kenny-ego poprzez nakładkę cv2.Canny z atrybutami low_threshold=20 oraz high_threshold=100, a następnie zapisujemy.

```
1 usage   jerzy
def canny(img_path, new_img_path):
    canny_low_threshold = 20
    canny_high_threshold = 100
    gray_blur = cv2.imread(img_path)
    gray_blur_canny = cv2.Canny(gray_blur, canny_low_threshold, canny_high_threshold)
    cv2.imshow( winname: 'Grayscale blur canny', gray_blur_canny)
    cv2.imwrite(new_img_path, gray_blur_canny)
```

Kadrowanie

Z obrazka z zaznaczonymi konturami wybieramy interesujący nas obszar poprzez ustawienie odpowiednich wartości h, w, x, y I następnie zapisujemy wycięty fragment:

```
1 usage  jerzy
def kenny(img_path, new_img_path):
    img = cv2.imread(img_path, flags: 0)
    height, width = img.shape[:2]
    print(height, width)
    h, w, x, y = 200, 1247, 400, 0
    img1 = img[x:x + h, y:y + w]
    img2 = np.zeros_like(img)
    img2[x:x + h, y:y + w] = img1
    cv2.imshow( winname: 'Kenny', img2)
    cv2.imwrite(new_img_path, img2)
```



Wyznaczanie linii prostych

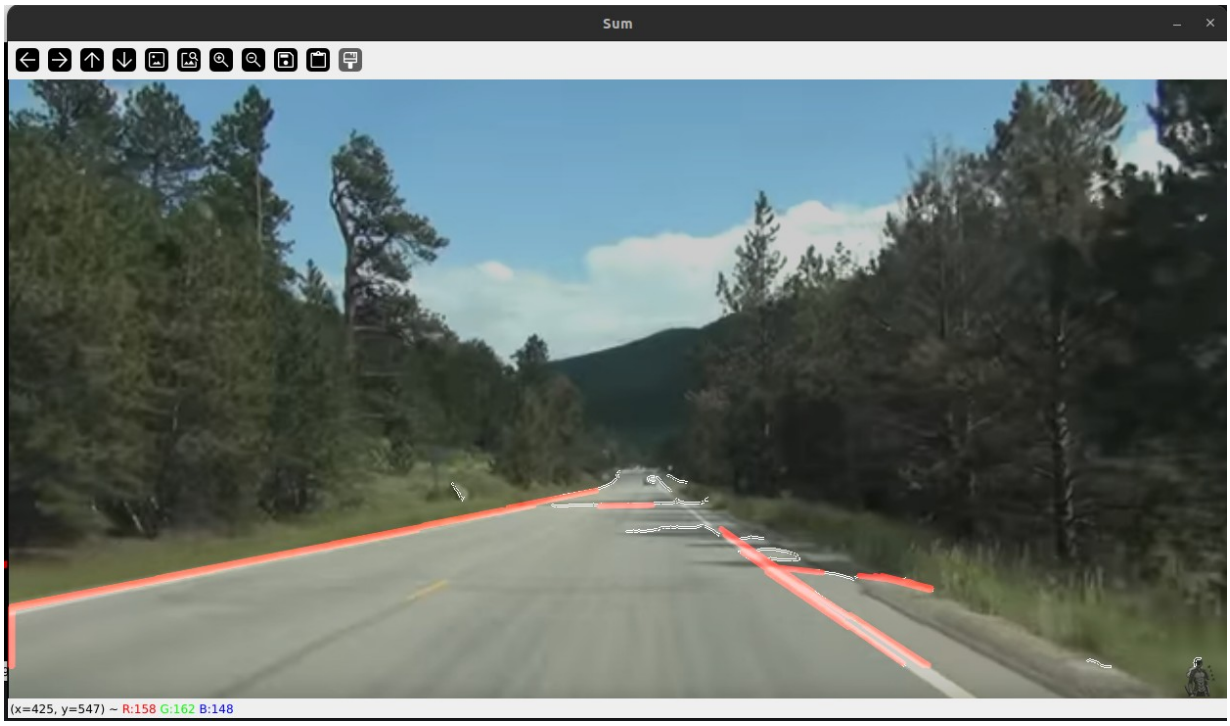
Linie proste wyznaczamy przy użyciu algorytmu Huffa. Linie proste zaznaczamy na nowym obrazku czerwonym kolorem I go zapisujemy:

```python
def huff(img_path, new_img_path):
    src = cv2.imread(img_path)
    dst = cv2.Canny(src, 50, 200, None, 3)
    cdst = cv2.cvtColor(dst, cv2.COLOR_GRAY2BGR)
    cdstP = np.copy(cdst)
    lines = cv2.HoughLines(dst, rho: 1, np.pi / 100, threshold: 150, lines: None, srn: 0, stn: 0)
    if lines is not None:
        for i in range(0, len(lines)):
            rho = lines[i][0][0]
            theta = lines[i][0][1]
            a = math.cos(theta)
            b = math.sin(theta)
            x0 = a * rho
            y0 = b * rho
            pt1 = (int(x0 + 1000 * (-b)), int(y0 + 1000 * a))
            pt2 = (int(x0 - 1000 * (-b)), int(y0 - 1000 * a))
            cv2.line(cdst, pt1, pt2, color: (0, 0, 255), thickness: 3, cv2.LINE_AA)
    linesP = cv2.HoughLinesP(dst, rho: 1, np.pi / 100, threshold: 50, lines: None, minLineLength: 50, maxLineGap: 10)
    if linesP is not None:
        for i in range(0, len(linesP)):
            l = linesP[i][0]
            cv2.line(cdstP, pt1: (l[0], l[1]), pt2: (l[2], l[3]), color: (0, 0, 255), thickness: 3, cv2.LINE_AA)
    cv2.imshow( winname: 'Detected lines (in red)', cdstP)
    cv2.imwrite(new_img_path, cdstP)
```

Naniesienie na oryginalny obraz

Na końcu pozostało nanieść zaznaczone linie proste na oryginalny obraz poprzez nakładkę cv2.addWeighted, w której podajemy oryginalny obraz, parametr alpha=0.8, obraz z czerwonymi liniami, beta=1 oraz gamma=0. Utworzony obraz zapisujemy:

```
1 usage   jerzy
def put_on_original(img_path, original_img_path, new_img_path):
    img = cv2.imread(original_img_path)
    img1 = cv2.imread(img_path)
    img2 = cv2.addWeighted(img, alpha: 0.8, img1, beta: 1, gamma: 0)
    cv2.imshow( winname: 'Sum', img2)
    cv2.imwrite(new_img_path, img2)
```



Uruchomienie programu

Wywołujemy wszystkie funkcje programu po kolei:

```
image = 'droga.png'
convert_to_gray(image, new_img_path: 'droga_gray.png')
gauss( img_path: 'droga_gray.png', new_img_path: 'droga_gray_blur.png')
canny( img_path: 'droga_gray_blur.png', new_img_path: 'droga_gray_blur_canny.png')
kenny( img_path: 'droga_gray_blur_canny.png', new_img_path: 'droga_gray_blur_canny_kenny.png')
huff( img_path: 'droga_gray_blur_canny_kenny.png', new_img_path: 'huff.png')
put_on_original( img_path: 'huff.png', original_img_path: 'droga.png', new_img_path: 'result.png')
cv2.waitKey()
cv2.destroyAllWindows()
```