

Zagovor laboratorijskih vaj - Naloga 2021-3

Pripravila: Gašper Podobnik & Tomaž Vrtovec

Naloga

Z osnovnim znanjem iz obdelave slik lahko na slikah najdemo enostavne vzorce. Pri tej nalogi boste s pomočjo Houghove preslikave detektirali in orisali kvadrat, ki je narisana na zemljevidu Pariza in označuje ulice okoli parka Palais-Royal:

Vhodna slika



Dana je dvodimenzionalna (2D) RGB slika `paris_map-807-421.png` velikosti $X \times Y = 807 \times 421$ slikovnih elementov, ki je zapisana v PNG formatu, pri čemer so slikovni elementi izotropni.

1. Napišite funkcijo za pretvorbo barvne slike v sivinsko sliko:

```
def color2grayscale(iImage):  
    # ...  
    return oImage
```

kjer vhodni argument `iImage` predstavlja barvno RGB sliko, izhodni argument `oImage` pa predstavlja 2D sivinsko sliko. Vrednosti sivinske slike naj bodo povprečje vrednosti R, G in B komponent barvne slike, zaokrožene na celo število navzdol.

Intenzitete vhodne slike najprej preslikajte tako, da bodo ležale na intervalu $[0, 255]$. Prikažite rezultat pretvorbe vhodne slike v sivinsko sliko.

2. Da bo naloga detekcije kvadrata na sliki enostavnejša, poravnajte sivinsko sliko dobljeno v prvi nalogi tako, da bodo stranice kvadrata vzporedne s koordinatnimi osmi, kot je prikazano na sliki spodaj. Pomagajte si lahko z naslednjimi koraki:

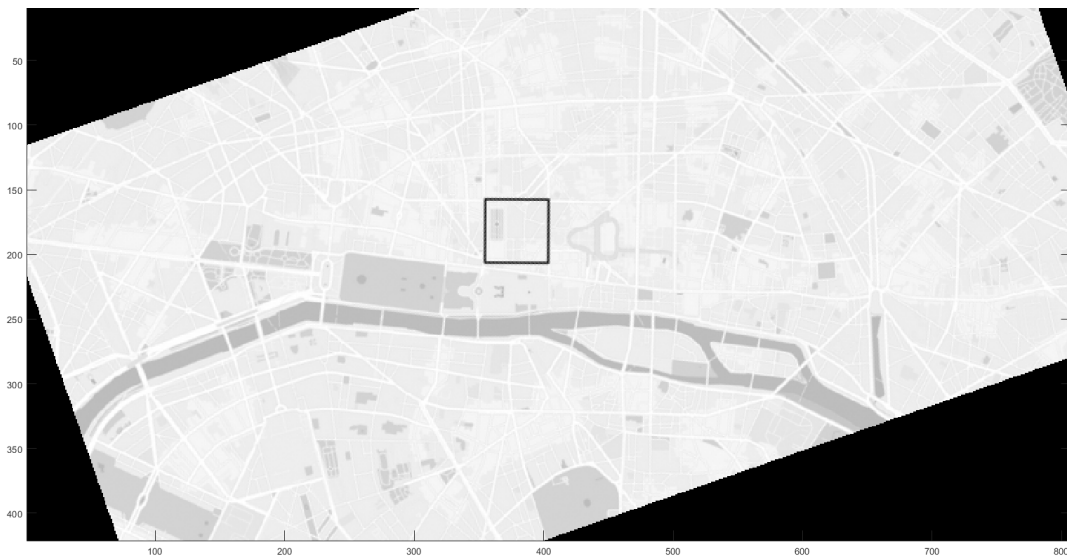
- določite točko *A*, ki leži v zgornjem levem oglišču kvadrata, in točko *B*, ki leži v zgornjem desnem oglišču kvadrata, glej sliko.

Namig: koordinati obeh točk lahko na primer odčitane s pomočjo kurzorja v Matlabovem grafičnem vmesniku za prikaz slik.

- izračunajte kót ϕ med vektorjema \vec{a} in \vec{b} . Vektor \vec{a} povezuje točki A in B , za vektor \vec{b} pa lahko izberete enotski vektor v smeri osi x . Velja enačba skalarnega produkta vektorjev $\vec{a} \cdot \vec{b} = |\vec{a}| |\vec{b}| \cos(\phi)$.
- dobljeni kót uporabite za rotacijo sivinske slike okoli točke A . Uporabite linearno interpolacijo, vrednost ozadja nastavite na 0.

Namig: pomagati si lahko s funkcijo `transformImage` iz laboratorijskih vaj, ki jo je potrebno nadgraditi tako, da omogoča rotacijo okoli želene točke.

Poravnana slika



3. Iz poravnane sivinske slike, ki ste jo pripravili v nalogi 2, izločite robove. Uporabite lahko na primer Sobelov operator, pri tem pa poskrbite, da bodo intenzitete na amplitudni sliki gradienta ležale na intervalu $[0, 255]$.

Prikažite sliko robov, ki jo dobite po upravljanju vrednosti. Prag izberite smiselno, t.j. tako da bodo robovi kvadrata dobro vidni.

4. Napišite funkcijo, ki bo na podlagi Houghove preslikave določila koordinate središča kvadrata z dano dolžino stranice:

```
def getSquareCenterPoint(iImage, iLength):
    # ...
    return oCenter, oAcc
```

kjer vhodni argument `iImage` predstavlja binarno sliko robov, `iLength` pa dolžino stranice kvadrata, ki jo nastavite tako, da bo enaka dolžini stranice kvadrata, ki ga želimo detektirati. Izhodni argument `oCenter` predstavlja koordinate središča kvadrata (x_0, y_0) , `oAcc` pa akumulator Houghove preslikave.

V pomoč vam je lahko funkcija `getCenterPoint`, ki ste jo pripravili v okviru domače naloge pri vaji z naslovom *Houghova preslikava*.

Prikažite središče in stranice kvadrata na poravnani sivinski sliki, kot je prikazano na sliki spodaj. Poleg tega prikažite še sliko akumulatorja Houghove preslikave.

Konční rezultat

