

## Vaja 6: Geometrijske preslikave slik

Priprava: Gašper Podobnik & Tomaž Vrtovec

### Navodila

Geometrijska preslikava  $\mathcal{T}: \mathbb{R}^n \rightarrow \mathbb{R}^n$  omogoča preslikavo vseh slikovnih elementov slike iz lokacij  $(x, y, z, \dots)$  na nove lokacije  $(u, v, w, \dots) = \mathcal{T}(x, y, z, \dots)$ . Najbolj splošna linearna preslikava je *afina preslikava*, med nelinearnimi preslikavami pa je zelo uveljavljena *radialna preslikava*.

1. Dana je dvodimenzionalna (2D) sivinska slika `lena-256x512-08bit.raw` velikosti  $X \times Y = 256 \times 512$  slikovnih elementov, ki je zapisana v obliki surovih podatkov (RAW) z 8 biti na slikovni element, velikost slikovnega elementa pa je enaka  $\Delta x \times \Delta y = 2,0 \times 1,0$  mm. Naložite in prikažite dano sliko.

2. Napišite funkcijo za izračun parametrov afine oz. radialne preslikave v 2D:

```
def getParameters(iType, **kwargs):  
    # ...  
    # your code goes here  
    # ...  
    return oP
```

kjer vhodni argument `iType` predstavlja vrsto preslikave (`'affine'` za afino, `'radial'` za radialno), `kwargs` pa večje število vhodnih argumentov glede na vrsto preslikave, in sicer:

- za *afino preslikavo*: vektor skaliranja oz. povečave  $[k_x, k_y] = (k_x, k_y)$ , vektor translacije oz. premika  $[t_x, t_y] = (t_x, t_y)$ , kot rotacije oz. vrtenja  $\phi = \varphi$  in vektor striga oz. zatega  $[g_x, g_y] = (g_{xy}, g_{yx})$ ;
- za *radialno preslikavo*: matrika kontrolnih točk  $[x_k, y_k] = (x_k, y_k)$  in matrika preslikanih kontrolnih točk  $[u_k, v_k] = (u_k, v_k)$ , pri čemer  $k$ -ta vrstica obeh matrik predstavlja  $k$ -to od skupno  $K$  kontrolnih točk ( $k = 1, 2, \dots, K$ ).

Izhodni argument `oP` predstavlja parametre preslikave glede na vrsto preslikave, in sicer:

- za *afino preslikavo*: matrika afine preslikave  $\mathbf{T} = \mathbf{T}_g \cdot \mathbf{T}_\varphi \cdot \mathbf{T}_t \cdot \mathbf{T}_k$ , ki je sestavljena iz matrik štirih elementarnih linearnih preslikav, kjer je  $\mathbf{T}_k$  matrika skaliranja,  $\mathbf{T}_t$  matrika translacije,  $\mathbf{T}_\varphi$  matrika rotacije in  $\mathbf{T}_g$  matrika striga:

$$\mathbf{T}_k = \begin{bmatrix} k_x & 0 & 0 \\ 0 & k_y & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad \mathbf{T}_t = \begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix}, \quad \mathbf{T}_\varphi = \begin{bmatrix} \cos \varphi & -\sin \varphi & 0 \\ \sin \varphi & \cos \varphi & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad \mathbf{T}_g = \begin{bmatrix} 1 & g_{xy} & 0 \\ g_{yx} & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix};$$

- za *radialno preslikavo*: matrika kontrolnih točk  $[x_k, y_k] = (x_k, y_k)$  in matrika koeficientov pripadajočih radialnih funkcij  $[A_k, B_k] = (\alpha_k, \beta_k)$  (izhodni argument naj bo v obliki Pythonovega slovarja, kjer vsaka izmed izhodnih matrik predstavlja eno vrednost), pri čemer  $k$ -ta vrstica obeh matrik predstavlja  $k$ -to od skupno  $K$  kontrolnih točk ( $k = 1, 2, \dots, K$ ), tako da velja:

$$u_i = \sum_{k=1}^K \alpha_k U\left(\|(x_i, y_i) - (x_k, y_k)\|\right), \quad v_i = \sum_{k=1}^K \beta_k U\left(\|(x_i, y_i) - (x_k, y_k)\|\right),$$

kjer  $r = \|(x_i, y_i) - (x_k, y_k)\|$  predstavlja razdaljo med kontrolno točko  $(x_i, y_i)$  in  $k$ -to kontrolno točko  $(x_k, y_k)$ ,  $U(r) = -r^2 \ln r$  pa radialno funkcijo, ki opisuje ukrivljanje tanke plošče.

Izračunajte parametre preslikav z naslednjimi vrednostmi vhodnih argumentov:

- za *afino preslikavo*:  $\varphi = 30^\circ$ , ostale parametre pa nastavite tako, da se pripadajoče elementarne preslikave ne izvedejo;
- za *radialno preslikavo*: kontrolne točke  $(x_1, y_1) = (0, 0)$  mm,  $(x_2, y_2) = (511, 0)$  mm,  $(x_3, y_3) = (0, 511)$  mm in  $(x_4, y_4) = (511, 511)$  mm ter pripadajoče preslikane kontrolne točke  $(u_1, v_1) = (0, 0)$  mm,  $(u_2, v_2) = (511, 0)$  mm,  $(u_3, v_3) = (0, 511)$  mm in  $(u_4, v_4) = (255, 255)$  mm.

### 3. Napišite funkcijo za afino oz. radialno preslikavo slike v 2D:

```
def transformImage(iType, iImage, iDim, iP, iBgr=0, iInterp=0):
    # ...
    # your code goes here
    # ...
    return oImage
```

kjer vhodni argument `iType` predstavlja vrsto preslikave (`'affine'` za afino, `'radial'` za radialno), `iImage` vhodno sliko preslikave, `iDim`  $(\Delta x, \Delta y)$  velikost slikovnega elementa, `iP` parametre preslikave (izhodni argument funkcije `getParameters()`), `iBgr` vrednost slikovnih elementov v preslikani sliki, ki nimajo pripadajočega para v vhodni sliki (t.i. sivinska vrednost ozadja slike), `iInterp` pa red interpolacije. Privzeta vrednost za `iBgr` naj bo 0, za `iInterp` pa tudi 0, torej interpolacija ničtega reda. Izhodni argument `oImage` predstavlja preslikano sliko, tako da se je na lokacijo  $(u, v)$  preslikal slikovni element iz lokacije  $(x, y)$  vhodne slike:

- za *afino preslikavo*:

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \mathbf{T} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix};$$

- za *radialno preslikavo*:

$$u = \sum_{k=1}^K \alpha_k U\left(\|(x, y) - (x_k, y_k)\|\right), \quad v = \sum_{k=1}^K \beta_k U\left(\|(x, y) - (x_k, y_k)\|\right).$$

V obeh primerih lahko za določanje slikovnih vrednosti uporabite interpolacijo ničtega reda (po principu najbližjega soseda).

Izvedite afino in radialno preslikavo nad dano sliko na podlagi parametrov iz točke 2, pri čemer za sivinsko vrednost ozadja slike izberete 63. Prikažite preslikane slike.

## Vprašanja

Odgovore na sledeča vprašanja zapišite v poročilo, v katerega vstavite zahtevane izrise in programske kode.

Poleg slike `lena-256x512-08bit.raw` je dana tudi slika `grid-256x512-08bit.raw`, ki ima enako velikost, obliko zapisa in velikost slikovnega elementa kot slika `lena-256x512-08bit.raw`, služi pa za lažjo predstavitev učinkov izvedenih geometrijskih preslikav. Poleg tega pri vseh spodaj podanih problemih za sivinsko vrednost ozadja slike izberete 63.

1. Spremenite funkcijo `transformImage()` tako, da bo za določanje sivinskih vrednosti v preslikani sliki poleg interpolacije ničtega reda omogočala tudi interpolacijo prvega reda. Izvedite afino preslikavo nad sliko `grid-256x512-08bit.raw` s parametroma  $k_y = 0,8$  in  $g_{xy} = 0,5$  (preostali parametri naj ne vplivajo na preslikavo), pri čemer za določanje sivinskih vrednosti enkrat uporabite interpolacijo ničtega reda, drugič pa interpolacijo prvega reda.

Priložite programsko kodo spremenjene funkcije `transformImage()` ter izrisa slik po preslikavi.

2. Izvedite vsako od naslednjih afinih preslikav nad sliko `lena-256x512-08bit.raw`, pri čemer uporabite samo podane parametre (preostali parametri naj ne vplivajo na preslikavo) ter interpolacijo sivinskih vrednosti prvega reda:

- (a)  $k_x = 0,7$  in  $k_y = 1,4$ ;
- (b)  $t_x = 20$  mm in  $t_y = -30$  mm;
- (c)  $\varphi = -30^\circ$ ;
- (d)  $g_{xy} = 0,1$  in  $g_{yx} = 0,5$ ;
- (e)  $t_x = -10$  mm,  $t_y = 20$  mm in  $\varphi = 15^\circ$ ;
- (f)  $k_x = 0,7$ ,  $k_y = 0,7$ ,  $t_x = 30$  mm,  $t_y = -20$  mm in  $\varphi = -15^\circ$ .

Priložite izrise slik za vsako preslikavo.

3. Kako se imenuje preslikava iz vprašanja 2(e) in kako preslikava iz vprašanja 2(f)? Opišite lastnosti teh preslikav.
4. Izvedite afini preslikavi iz vprašanja 2(c) in vprašanja 2(d) nad sliko `lena-256x512-08bit.raw`, pri čemer izhodišče koordinatnega sistema preslikave prestavite v središče slike (tako da se slika npr. vrti okoli svojega središča).

Priložite izrise slik za vsako preslikavo ter programsko kodo, s katero ste dosegli prestavitev koordinatnega sistema preslikave.

5. Izvedite radialno preslikavo z naslednjimi kontrolnimi točkami  $(x_k, y_k)$ :

$$\begin{aligned}(x_1, y_1) &= (0, 0) \text{ mm}, & (x_2, y_2) &= (511, 0) \text{ mm}, & (x_3, y_3) &= (0, 511) \text{ mm}, \\(x_4, y_4) &= (511, 511) \text{ mm}, & (x_5, y_5) &= (63, 63) \text{ mm}, & (x_6, y_6) &= (63, 447) \text{ mm}, \\(x_7, y_7) &= (447, 63) \text{ mm}, & (x_8, y_8) &= (447, 447) \text{ mm},\end{aligned}$$

ter pripadajočimi preslikanimi kontrolnimi točkami  $(u_k, v_k)$ :

$$\begin{aligned}(u_1, v_1) &= (0, 0) \text{ mm}, & (u_2, v_2) &= (511, 0) \text{ mm}, & (u_3, v_3) &= (0, 511) \text{ mm}, \\(u_4, v_4) &= (511, 511) \text{ mm}, & (u_5, v_5) &= (127, 95) \text{ mm}, & (u_6, v_6) &= (127, 415) \text{ mm}, \\(u_7, v_7) &= (383, 95) \text{ mm}, & (u_8, v_8) &= (383, 415) \text{ mm}.\end{aligned}$$

Na vhodno in preslikano sliko narišite kontrolne točke z križci rdeče barve in preslikane kontrolne točke z krožci modre barve, kar storite z uporabo naslednje funkcije neposredno po klicu funkcije `displayImage()`:

```
def displayPoints(iXY, iMarker):  
    plt.plot(iXY[:,0], iXY[:,1], iMarker, ms=10, lw=2)
```

kjer vhodni argument `iXY` predstavlja matriko točk  $[x_j, y_j] = (x_j, y_j)$  ( $j$ -ta vrstica matrike predstavlja  $j$ -to od skupno  $J$  točk), `iMarker` pa barvo in vrsto izrisa točk (npr. `'rx'` za rdeče križce, `'bo'` za modre krožce). Da bo izris deloval pravilno je potrebno 'zakomentirati' ukaz `plt.show()` na koncu funkcije `displayImage()`.

Priložite izrise originalne in preslikane slike z vrisanimi kontrolnimi in preslikanimi kontrolnimi točkami, in sicer za radialno preslikavo nad sliko `grid-256x512-08bit.raw` ter za radialno preslikavo nad sliko `lena-256x512-08bit.raw`.

Ali glede na položaj točk preslikava deluje pravilno? Obrazložite odgovor.

## Dodatek

Odgovore na sledeče probleme ni potrebno prilagati k poročilu, prispevajo pa naj k boljšemu razumevanju vsebine.

Ena izmed razširjenih nelinearnih preslikav je tudi *polinomska preslikava*, ki vsak slikovni element na lokaciji  $(x, y)$  na vhodni sliki preslika v lokacijo  $(u, v)$  v skladu z naslednjimi pravili:

$$u = \sum_{i=0}^n \sum_{j=0}^i a_{i-j,j} x^{i-j} y^j, \quad v = \sum_{i=0}^n \sum_{j=0}^i b_{i-j,j} x^{i-j} y^j,$$

kjer ob izbrani stopnji polinoma  $n$  in določenih kontrolnih točkah najprej izračunamo koeficiente polinomov  $a_{i-j,j}$  in  $b_{i-j,j}$ , nato pa le-te uporabimo v sami preslikavi.

Spremenite funkciji `getParameters()` in `transformImage()` tako, da bosta za vhodni argument `iType = 'polynomial'` izvedli tudi polinomsko preslikavo.

