

# Procesarea Imaginilor Digitale

## Tema 7 - Transformări geometrice

1. Scalarea imaginilor folosind pentru interpolare cel mai apropiat vecin(**1p**)

### Cerințe

- adăugați o opțiune de meniu - *Scalare*
- citiți dintr-o casetă de dialog coeficientul de scalare (poate fi și mai mic decât 1, de exemplu 0.5).
- aplicați operația de scalare pornind de la imaginea rezultat către imaginea sursă și alegând pixelul cel mai apropiat de coordonatele reale din imaginea sursă obținute prin transformarea inversă. Pentru imagini RGB realizați aceste operații pe fiecare canal în parte.

2. Rotația imaginilor folosind pentru interpolare cel mai apropiat vecin(**1p**)

### Cerințe

- adăugați o opțiune de meniu - *Rotatie*
- citiți dintr-o casetă de dialog unghiul de rotație (în grade) (Obs. Trebuie transformat în radiani).
- aplicați operația de rotație pornind de la imaginea rezultat către imaginea sursă și alegând pixelul cel mai apropiat de coordonatele reale din imaginea sursă obținute prin transformarea inversă. Pentru imagini RGB realizați aceste operații pe fiecare canal în parte.

3. Scalarea imaginilor folosind interpolarea biliniară. (**2p**)

### Cerințe

- adăugați o opțiune de meniu - *Scalare*
- citiți dintr-o casetă de dialog coeficientul de scalare (poate fi și mai mic decât 1, de exemplu 0.5).
- aplicați operația de scalare pornind de la imaginea rezultat către imaginea sursă și folosind interpolarea biliniară. Pentru imagini RGB realizați aceste operații pe fiecare canal în parte.

4. Scalarea imaginilor folosind interpolarea biliniară. **(3p)**

**Cerințe**

- adăugați o opțiune de meniu - *Scalare*
- citiți dintr-o casetă de dialog coeficientul de scalare (poate fi și mai mic decât 1, de exemplu 0.5).
- aplicați operația de scalare pornind de la imaginea rezultat către imaginea sursă și folosind interpolarea biliniară. Pentru imagini RGB realizați aceste operații transformând întâi în HSV și apoi realizând pe fiecare canal HSV în parte. Retransformați la loc.

5. Rotația imaginilor folosind interpolarea biliniară. **(2p)**

**Cerințe**

- adăugați o opțiune de meniu - *Rotatie*
- citiți dintr-o casetă de dialog unghiul de rotație (în grade) (Obs. Trebuie transformat în radiani).
- aplicați operația de rotație pornind de la imaginea rezultat către imaginea sursă și folosind interpolarea biliniară. Pentru imagini RGB realizați aceste operații pe fiecare canal în parte.

6. Rotația imaginilor folosind interpolarea biliniară. **(2p)**

**Cerințe**

- adăugați o opțiune de meniu - *Rotatie*
- citiți dintr-o casetă de dialog unghiul de rotație (în grade) (Obs. Trebuie transformat în radiani).
- aplicați operația de rotație pornind de la imaginea rezultat către imaginea sursă și folosind interpolarea biliniară. Pentru imagini RGB realizați aceste operații transformând întâi în HSV și apoi realizând pe fiecare canal HSV în parte. Retransformați la loc.

7. *Twirl Transform* **(2p)**

Pentru această transformare funcția inversă (cea care se aplică de la imaginea rezultat către imaginea sursă) este dată prin:

$$T_x^{-1} : x = \begin{cases} x_0 + r \cos \beta, & r \leq r_{max} \\ x', & r > r_{max} \end{cases}$$
$$T_y^{-1} : y = \begin{cases} y_0 + r \sin \beta, & r \leq r_{max} \\ y', & r > r_{max} \end{cases}$$

unde,  $d_x = x' - x_0$ ,  $d_y = y' - y_0$ ,  $(x_0, y_0)$  - centrul de rotație, poate fi considerat centrul imaginii,  $r = \sqrt{d_x^2 + d_y^2}$  = distanța de la centrul de rotație la punctul  $(x', y')$  din imaginea rezultat,  $\beta = \text{atan}_2(d_y, d_x) + \alpha((r_{max} - r)/r_{max})$ .

- adăugați o opțiune de meniu - *Twirl*
- citiți dintr-o casetă de dialog unghiul de rotație  $\alpha$  (în grade) (Obs. Trebuie transformat în radiani) și raza maximă  $r_{max}$ .
- aplicați operația de *twirl* pornind de la imaginea rezultat către imaginea sursă și alegând pixelul cel mai apropiat de coordonatele reale din imaginea sursă obținute prin transformarea inversă descrisă de formulele de mai sus și folosind interpolarea biliniară. Pentru imagini RGB realizați aceste operații pe fiecare canal în parte.

#### 8. *Ripple Transform* (2p)

Pentru această transformare funcția inversă (cea care se aplică de la imaginea rezultat către imaginea sursă) este dată prin:

$$T_x^{-1} : x = x' + a_x \sin\left(\frac{2\pi y'}{t_x}\right)$$

$$T_y^{-1} : y = y' + a_y \sin\left(\frac{2\pi x'}{t_y}\right)$$

unde,  $t_x$ ,  $t_y$ ,  $a_x$ ,  $a_y$  - parametri care determină gradul de deformare și care se citesc dintr-o casetă de dialog. Valori de exemplificare:  $t_x = 120$ ,  $t_y = 250$ ,  $a_x = 10$ ,  $a_y = 15$ .

- adăugați o opțiune de meniu - *Ripple*
- citiți dintr-o casetă de dialog parametri  $t_x$ ,  $t_y$ ,  $a_x$ ,  $a_y$
- aplicați operația de *ripple* pornind de la imaginea rezultat către imaginea sursă și alegând pixelul cel mai apropiat de coordonatele reale din imaginea sursă obținute prin transformarea inversă descrisă de formulele de mai sus și folosind interpolarea biliniară. Pentru imagini RGB realizați aceste operații pe fiecare canal în parte.

#### 9. Deformare sferică (3p)

Pentru această transformare funcția inversă (cea care se aplică de la imaginea rezultat către imaginea sursă) este dată prin:

$$T_x^{-1} : x = x' - \begin{cases} z \tan(\beta_x), & r \leq r_{max} \\ 0, & r > r_{max} \end{cases}$$

$$T_y^{-1} : y = y' - \begin{cases} z \tan(\beta_y), & r \leq r_{max} \\ 0, & r > r_{max} \end{cases}$$

unde,  $d_x = x' - x_0$ ,  $d_y = y' - y_0$ ,  $(x_0, y_0)$  - centrul de deformare, poate fi considerat centrul imaginii  $r = \sqrt{d_x^2 + d_y^2}$  = distanța de la  $(x_0, y_0)$  la punctul  $(x', y')$ ,  $z = \sqrt{r_{max}^2 - r^2}$ ,  $\beta_x = \left(1 - \frac{1}{\rho}\right) \sin^{-1} \left(\frac{d_x}{\sqrt{d_x^2 + z^2}}\right)$ ,  $\beta_y = \left(1 - \frac{1}{\rho}\right) \sin^{-1} \left(\frac{d_y}{\sqrt{d_y^2 + z^2}}\right)$ .  $r_{max}$  = raza lentilei și  $\rho$  = indicele de refracție, se citesc dintr-o casetă de dialog. Exemple de valori pentru testare:  $r_{max}$  = jumătate din lățimea imaginii,  $\rho = 1.8$  și folosind interpolarea biliniară.

- adăugați o opțiune de meniu - *Deformare sferică*
- citiți dintr-o casetă de dialog parametri  $\rho$  și  $r_{max}$
- aplicați operația de deformare sferică pornind de la imaginea rezultat către imaginea sursă și alegând pixelul cel mai apropiat de coordonatele reale din imaginea sursă obținute prin transformarea inversă descrisă de formulele de mai sus. Pentru imagini RGB realizați aceste operații pe fiecare canal în parte.

10. Scalarea imaginilor folosind interpolarea bicubică. **(3p)**

**Cerințe**

- adăugați o opțiune de meniu - *Scalare*
- citiți dintr-o casetă de dialog coeficientul de scalare (poate fi și mai mic decât 1, de exemplu 0.5).
- aplicați operația de scalare pornind de la imaginea rezultat către imaginea sursă și folosind interpolarea bicubică. Pentru imagini RGB realizați aceste operații pe fiecare canal în parte.

11. Rotația imaginilor folosind interpolarea bicubică. **(3p)**

**Cerințe**

- adăugați o opțiune de meniu - *Rotatie*
- citiți dintr-o casetă de dialog unghiul de rotație (in grade) (Obs. Trebuie transformat în radiani).
- aplicați operația de rotație pornind de la imaginea rezultat către imaginea sursă și folosind interpolarea bicubică. Pentru imagini RGB realizați aceste operații pe fiecare canal în parte.

12. Transformarea proiectivă. **(4p)**- interpolare biliniară / **(5p)** - interpolare bicubică)

**Cerințe**

- adăugați o opțiune de meniu - *Transformare proiectivă*
- alegeți cele patru puncte din imaginea sursă prin click de mouse și setați un dreptunghi corespunzător în imaginea rezultat.
- calculați matricea de transformare.
- transformați imaginea sursă folosind interpolarea biliniară / bicubică. Pentru imagini RGB realizați aceste operații pe fiecare canal în parte.