

Zagovor laboratorijskih vaj - Naloga 2022-2

Pripravili: Luka Škrlić, Gašper Podobnik & Tomaž Vrtovec

Naloga

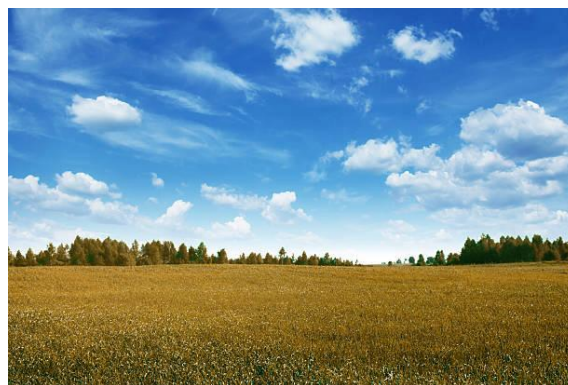
RGB barvni model sliko predstavi s tremi kanali; kanalom za rdečo, zeleno in modro barvno komponento. Kot alternative temu načinu zapisa so se uveljavili še drugi barvni modeli, npr. CMYK, HSL, YUV. Danes boste spoznali barvni model HSV, kjer prvi kanal predstavlja odtenek barve (ang. *hue*), drugi intenzivnost (ang. *saturation*), tretji pa vrednost oz. svetlost (ang. *value*).

Vaša naloga bo preslikava intenzitet iz modela RGB v model HSV, kjer boste opravili filtriranje. Dobljeno sliko boste nato preslikali nazaj v model RGB.

Vhodna RGB slika



Izhodna RGB slika



Dana je dvodimenzionalna (2D) barvna (RGB) slika `travnik-uint8.jpeg` velikosti $X \times Y \times RGB_{dim} = 612 \times 408 \times 3$ slikovnih elementov, ki je zapisana v formatu jpeg z 8 biti na slikovni element. Slikovni elementi so izotropni.

1. Naložite dano sliko in napišite funkcijo za normalizacijo intenzitet slike, tj. funkcijo, ki intenzitete vhodne slike s poljubnega intervala $[I_{min}, I_{max}]$ preslika na interval $[0, 1]$:

```
function oImage = normalizeImage(iImage),
```

kjer vhodni argument `iImage` predstavlja vhodno RGB sliko. Izhodni argument `oImage` je matrika velikosti $X \times Y \times 3$ (X in Y sta dimenziji vhodne slike), katere intenzitete ležijo na intervalu $[0, 1]$. *Namig:* pred normalizacijo ustrezno nastavite podatkovni tip slike.

Vhodno sliko normalizirajte in rezultat shranite v spremenljivko `img_normalized`. Dobljeno sliko prikažite.

2. Napišite funkcijo za pretvorbo slike iz barvnega modela RGB v HSV:

```
function oHSV = rgb2hsv(iRGB),
```

kjer vhodni argument `iRGB` predstavlja vhodno RGB sliko. Izhodni argument `oHSV` pa je slika zapisana v HSV modelu. Pomagajte si z naslednjimi izračuni, pri čemer so r_i , g_i in b_i vrednosti rdeče, zelene in modre barvne komponente v poljubnem slikovnem elementu (x_i, y_i) :

$$\begin{aligned}v_i &= \max(r_i, g_i, b_i) \\c_i &= \max(r_i, g_i, b_i) - \min(r_i, g_i, b_i) \\l_i &= v_i - \frac{c_i}{2} \\h_i &= \begin{cases} 60^\circ \cdot \left(\frac{g_i - b_i}{c_i}\right), & v_i = r_i \\ 60^\circ \cdot \left(2 + \frac{b_i - r_i}{c_i}\right), & v_i = g_i \\ 60^\circ \cdot \left(4 + \frac{r_i - g_i}{c_i}\right), & v_i = b_i \\ 0, & c_i = 0 \end{cases} \\s_i &= \begin{cases} 0, & v_i = 0 \\ \frac{c_i}{v_i}, & \text{drugače} \end{cases}\end{aligned}$$

Dobljene vrednosti h_i , s_i , v_i , ki sestavljajo HSV zapis (*hue*, *saturation* in *value*), zložite v matriko enake velikosti kot je vhodna RGB slika, tako da bo imel vsak slikovni element (x_i, y_i) vrednosti (h_i, s_i, v_i) .

Preizkusite funkcijo na normalizirani vhodni sliki `img_normalized` in dobljeno sliko shranite v spremenljivko `img_hsv` ter jo prikažite.

3. Sledi korak, kjer spremenite barvo travnika v rumeno. Iz HSV slike `img_hsv` izločite rezino, ki ustreza kanalu H , in jo shranite v spremenljivko `h_slice`. Vrednosti v `h_slice`, ki so manjše od 100, delite s številom dva. Z dobljeno rezino sestavite novo HSV sliko (rezini S in V ostajata nespremenjeni), ki jo shranite v spremenljivko `img_hsv_transformed` in jo prikažite.
4. Napišite funkcijo za pretvorbo slike iz barvnega modela HSV v RGB:

```
function oRGB = hsv2rgb(iHSV),
```

kjer vhodni argument `iHSV` predstavlja vhodno HSV sliko, `oRGB` pa izhodno RGB sliko. Pomagajte si z naslednjimi izračuni, pri čemer so h_i , s_i in v_i vrednosti treh HSV komponent v poljubnem slikovnem elementu (x_i, y_i) , *mod* pa pomeni modulo (ostanek pri celoštevilskem

deljenju):

$$\begin{aligned}
c_i &= v_i \cdot s_i \\
\hat{h}_i &= \frac{h_i}{60^\circ} \\
x_i &= c_i \cdot (1 - |\hat{h}_i \bmod 2 - 1|) \\
(\hat{r}_i, \hat{g}_i, \hat{b}_i) &= \begin{cases} (c_i, x_i, 0), & 0 \leq \hat{h}_i < 1 \\ (x_i, c_i, 0), & 1 \leq \hat{h}_i < 2 \\ (0, c_i, x_i), & 2 \leq \hat{h}_i < 3 \\ (0, x_i, c_i), & 3 \leq \hat{h}_i < 4 \\ (x_i, 0, c_i), & 4 \leq \hat{h}_i < 5 \\ (c_i, 0, x_i), & 5 \leq \hat{h}_i < 6 \end{cases} \\
m_i &= v_i - c_i \\
(r_i, g_i, b_i) &= (\hat{r}_i + m_i, \hat{g}_i + m_i, \hat{b}_i + m_i)
\end{aligned}$$

Dobljene vrednosti r_i, g_i, b_i zložite v matriko enake velikosti kot je vhodna HSV slika, tako da bo imel vsak slikovni element (x_i, y_i) vrednosti (r_i, g_i, b_i) .

Preizkusite funkcijo na sliki `img_hsv` in `img_hsv_transformed`.