

# Vaja 1: Pravila in uvod v obdelavo slik z Matlabom

Janez Perš, Marija Ivanovska

Laboratorij za strojno inteligenco  
Fakulteta za elektrotehniko, Univerza v Ljubljani  
e-mail: janez.pers@fe.uni-lj.si, marija.ivanovska@fe.uni-lj.si

## Povzetek

Ta vaja je namenjena utrjevanju vaših spretnosti pri delu z Matlabom ter spoznavanju osnovnih konceptov, ki jih srečate pri obdelavi slik. Spoznali boste, kako so v Matlabu predstavljene slike, ter se naučili naslavljanja posamezne slikovne elemente. Naučili se boste prebrati sliko ali videoposnetek iz diska v Matlabov delovni prostor (workspace), ter ju po obdelavi shraniti nazaj na disk. Izvedli boste tudi nekaj preprostih operacij na nivoju slikovnih elementov.

## 1 Branje in pisanje slik ter video datotek

Najprej iz strežnika prenesite k sebi dve sliki in en video posnetek. To lahko storite z uporabo orodja `wget`, lahko pa uporabite tudi brskalnik Firefox:

```
!wget http://vision.fe.uni-lj.si/classes/RV/vaje/vaja1/lena.tiff
!wget http://vision.fe.uni-lj.si/classes/RV/vaje/vaja1/baboon.tiff
!wget http://vision.fe.uni-lj.si/classes/RV/vaje/vaja1/Hollywood2-t00427-rgb.avi
```

Še razlaga: klicaj pomeni v Matlabu klic zunanjega programa, `wget` pa je zunanji programček, ki ga uporabimo za prenos datotek po protokolu HTTP.

Sivinske slike so v Matlabu zapisane kot (2D) matrike slikovnih elementov, barvne slike pa kot (3D) polja slikovnih elementov. Za primer lahko poskusimo "ročno" sestaviti naslednjo sivinsko sliko dimenzije 3x3 slikovnih elementov:

```
A=[20,100,255;255,0,80;50,100,150]; A = repmat(A,[1,1,3]); image(uint8(A));
```

Vrstice slike ustrezajo vrsticam matrike, stolpci slike pa stolpcem matrike. Funkcija `repmat()` je potrebna zato, ker zna Matlab prikazovati samo barvne slike, kar pomeni, da moramo sivinsko sliko ustrezno "razširiti". Funkcija `image()` prikaže sliko v Matlabovem oknu, pretvorba v 8 bitni nepredznačeni format je pa potrebna zato, da zna Matlab sliko pravilno skalirati. Po izvršitvi gornje vrstice boste zagledali sliko, katere sredinski element (2,2) je popolnoma črn, saj mu je pripisana vrednost 0. Zgornji desni element ter element (2,1) bosta popolnoma bele barve, saj jima je pripisana vrednost 255.

Barvne slike so zapisane kot 3D polja, kjer prvi dve dimenziji (vrstica in stolpec) ustrezata vrstici oziroma stolpcu slike, tretja dimenzija (v tej dimenziji ima slika vedno točno tri elemente) pa predstavlja tri barvne ravnine, torej rdečo, zeleno in modro.

Sliko preberemo iz diska s pomočjo funkcije `imread`. Na primer, sliko Lene preberete iz diska in prikažete na naslednji način:

```
A=imread('lena.tiff');
image(A);
```

Če želite izvedeti o tej funkciji kaj več, lahko dobite obsežno dokumentacijo s pomočjo ukaza `doc imread`, ki ga vtipkate v komandno okno Matlaba. Manj obsežno dokumentacijo pa dobite s pomočjo klica `help imread`. V nadaljevanju vaj bomo zato navajali le imena funkcij ki jih boste uporabljali, na vas pa je, da s pomočjo dokumentacije ugotovite način, kako jih uporabiti.

Za branje video datotek imate dve možnosti. V starejših verzijah Matlaba uporabite funkcijo `aviread`, ki pa ne vrne neposredno RGB slike, ampak strukturo, ki vsebuje dve komponenti, `cdata` in `colormap`. RGB slike so zapisane v komponenti `cdata`. V novejših verzijah Matlaba je na voljo komponenta `VideoReader`. Primere uporabe najdete v dokumentaciji Matlaba (`doc VideoReader`).

Sedaj, ko ste spoznali kako poteka branje slik in video datotek, naložite sliko `lena.tiff` in izpišite vrednosti vseh treh (R,G,B) kanalov slikovnega elementa v vrstici 100 in stolpcu 350. Podobno naredite še za sliko `image(A)baboon.tiff`, in za slike z indeksi od 50 do 60 v video posnetku `Hollywood2-t00427-rgb.avi`.

**Namig:** RGB komponente omenjenega slikovnega elementa slike "lena" imajo vrednosti 231, 126 in 109.

## 2 Naloga 1.1: Zapisovanje slik in posnetkov - pretvorba v sivinski prostor (1 točka)

Naloženo sliko `lena.tiff` pretvorite v sivinsko, z uporabo naslednje transformacije (in ne z uporabo vgrajenih funkcij kot je na primer `rgb2gray`):

$$Y_{ij} = 0.299 \cdot R_{ij} + 0.587 \cdot G_{ij} + 0.114 \cdot B_{ij},$$

kjer je  $Y_{ij}$  sivinska vrednost slikovnega elementa v  $i$ -ti vrstici in  $j$ -tem stolpcu,  $R_{ij}$ ,  $G_{ij}$  in  $B_{ij}$  pa so vrednosti istoležečih slikovnih elementov v rdeči, zeleni in modri barvni ravnini. Rezultat pretvorbe naj bi bila matrika, ki jo ne boste mogli pravilno prikazati s pomočjo funkcije `image`, saj ta pravilno prikaže le tri kanalne (RGB) slike. Zato sivinsko 2D sliko z uporabo funkcije `repmat` razširite nazaj na 3 kanale in si razširjeno sliko  $A$  ogledajte z uporabo funkcije `image(uint8(A))`. Shranite jo na disk s pomočjo funkcije `imwrite`.

Postopek ponovite še za sliko `baboon.tiff` in za slike posnetka `Hollywood2-t00427-rgb.avi`. Obdelati morate vse slike posnetka! Za zapisovanje video posnetka v starejših verzijah Matlaba uporabite funkciji `avifile` in `addframe`, v novejših pa uporabite komponento `VideoWriter`. Primere uporabe najdete v dokumentaciji Matlaba (`doc VideoWriter`).

## 3 Naloga 1.2: Zapisovanje slik in posnetkov - pretvorba v RGB prostor (1 točka)

Sivinski slike in posnetek iz prejšnje naloga pobarvajte z uporabo naslednjega pravila:

$$barva = \begin{cases} zelena & Y \leq 100, \\ oranžna & 100 < Y \leq 200, \\ rdeča & Y > 200, \end{cases} \quad (1)$$

kjer je  $Y$  sivinska vrednost določenega slikovnega elementa. RGB vrednosti zelene, oranžne in rdeče barve določite sami. Pri tem si lahko pomagata z barvno paleto, ki jo najdete na naslednji povezavi:

[https://www.w3schools.com/colors/colors\\_picker.asp](https://www.w3schools.com/colors/colors_picker.asp).