

# Kamera

- pinhole model kamere
  - preprost
  - malo svetlobe, daljši čas osvetlitve
- leče
  - boljše optične lastnosti
  - popačenje, lom svetlobe
- digitalni senzor
  - digitalizacija
  - prostorsko vzorčenje porazdelitve svetlobe
  - časovno vzorčenje - fiksni časovni interval
  - kvantizacija na fiksni interval
  - DPI (1 inch = 2.54cm)

## Naloge

1. Kockasta škatla s velikostjo stranice 10cm z majhno odprtino na prednji strani deluje kot pinhole kamera. Usmerimo jo proti drevesu, ki je od kamere oddaljeno 14m. Kako velika je slika drevesa, ki nastane na zadnji strani škatle, če je drevo visoko 5m?

3.57cm

2. Sliko z dimenzijama 3000 in 4000 slikovnih elementov natisnemo na tiskalniku z ločljivostjo 300 DPI. Kakšna je velikost slike na papirju v metričnih enotah?

25.4cm x 33.86cm

3. S kamero z fokusno dolžino  $f = 60\text{mm}$  posnamemo sliko vertikalnega valja, ki je od kamere oddaljen 95m. Določi višino valja, če v digitalizirani obliki slika valja po višini zavzame 200 slikovnih elementov. Ločljivost tipala je 2500 DPI.

$y = 0.20320\text{cm}$ ,  $Y = 3.21\text{m}$

4. Kamero z fokusno dolžino  $f = 50\text{mm}$  med zajemom slike zarotiramo za eno stopinjo okoli navpične osi. Za koliko slikovnih elementov se zameglji slika, če ima tipalo ločljivost 2000 DPI.

- razdalja, ki jo opisuje rotacija okoli osi se spreminja z globino:

$$b = 2Z \sin \frac{\alpha}{2}$$

- premik v projekciji:

$$y = -f \frac{b}{Z} = -f \frac{2Z \sin(\alpha / 2)}{Z} = -f 2 \sin(\alpha / 2)$$

- poračunamo:  $y = 0.872\text{mm}$
- pretvorba v slikovne elemente:  $68.661 = 69$

# Zapis slik

- Poznamo več zapisov slik, primernih za različne scenarije uporabe
  - rasterske slike (JPEG, PNG, GIF, TIFF, BMP)
    - Kompresija: lossy, lossless
  - vektorske slike (SVG, WMF)
- Glave datotek, podpisi

## Naloge

1. Določite število bytov, ki so potrebni za zapis barvne slike z velikostjo 1024x786 slikovnih elementov brez uporabe kompresije. Vrednost v vsakem barvnem kanalu je zapisana z 8, 16, 14 bitov.

804864, 1609728, 1408512

2. Črnobela televizija prikazuje sliko z resolucijo 625 x 512 slikovnih elementov (8bit). Slike prikazuje s hitrostjo 25 slik na sekundo. Kakšen je najkrajši čas, v katerem lahko taka televizija prikaže vse možne unikatne slike?

$256^{320000} / 25 * 60 * 60 * 24 * 365 = (256^4 / 788400000) * 256^{319996} = 5.45 * 255^{319996}$   
let

# Intezitetna slika

## RGB

- Slika ki jo naložimo v Octave je barvna (3je kanali) ali sivinska (1 kanal)
- Kombinacije barv
- Kanali - prikaz posameznih kanalov

## Sivinska/intezitetna slika

- Pretvorba treh kanalov v en kanal
- Uporaba rgb2gray
- Rezultat projekcije modela

## Binarna slika

- Samo dve vrednosti (true, false)
- Ponavadi dobimo z uporabo primerjav in logičnih pravil

## Sivinski histogram

- poenostavitev zvezne verjetnostne porazdelitve
  - vizualizacija
  - lastnosti porazdelitve (ekstremi, itd.)
- sivinska slika - 1D histogram
- širina celice histograma

- prevelika - izgubimo informacije
- premajhna - premalo elementov v posamezni celici, vpliv naključnih variacij

## Koda

### grayscale.m

```
function [ g ] = grayscale(slika)
g = slika(:,:,1).*0.3 + slika(:,:,2).*0.59 + slika(:,:,3).*0.11;
```

---

### histogram.m

```
function [ h, celice ] = histogram(slika , nbins)

elementi = slika(:) + 1;
h = zeros(1,nbins); % inicializiramo histogram
f = 256 / (nbins - 1);

% pretvorba vrednosti elementov v indekse histograma
idx = ceil(elementi ./ f) + 1;

% možnost 1: povecajmo vsebino pripadajoce celice v histogramu
for i = idx'
    h(i) = h(i) + 1;
end

% možnost 2: nastavimo vsebino pripadajoce celice v histogramu
%for i = 1:nbins
%    h(i) = sum(idx == i);
%end

% normaliziraj histogram, da je vsota celic (integral) enaka 1
h = h / sum(h);
% izracunaj vhodno vrednost za celice histograma
celice = ((1 : nbins) - 1) .* f;
```

---

### backproject.m

```
function [ b ] = backproject(slika , h)

l = length(h) - 1;
ind = ceil((double(slika) + 1) ./ (256 / l)) + 1;
b = h(ind);
```

---

Inicializacija šumne slike:

```
- enakomerni:
uint8(rand(500, 500) * 255)

- normalna porazdelitev:
uint8(min(max(0, normrnd(128, 20, [500, 500])), 255)) % potrebujemo paket
statistics
```