**Spracovanie obrazu**

* Doteraz sme pracovali s 1D - so zvukom
* **Obrázky sú 2D - čiernobiela fotka - GRAYSCALE**
* Niekoľkokrát 2D - farebná fotka
* 3D - hĺbková mapa - Kinect - point clouds
* Video - 4D - čas

**Obrázok**

* Riadky indexované k, stĺpce indexované l, rozsah [0..k-1], [0..l-1]
* Jedna zložka je picture element = PIXEL
* Bežné kvantovanie 8/16 bitov [0..255], pre nás lepšie [0..1], 0 je čierna, 1 je biela

**Operácie nad pixelmi**

* Zvýšenie/zníženie jasu
* Zmena kontrastu
* Saturácia/klipovanie

**Histogram**

* Štatistiky hodnôt si nahodíme do grafu
* Všetky možné hodnoty pixelu nanesieme na 1 stranu, na 2 stranu dáme koľko hodnôt je pre každú farbu v obrázku
* **Ekvalizácia histogramu** - úprava tak, aby bolo lepšie vidieť jednotlivé výrazné hodnoty, pozor ale na stratu kvality obrázku
* **Prahovanie** - úprava na skutočne čiernobiely obrázok, určíme si prah, teda hranicu, kde sa to má predeliť

**Filtrácia signálu**

* Úlohou je získať kvalitnejší signál, alebo signál s požadovanými vlastnosťami

x[k,l]

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  | 9 | 8 | 7 |  |  |  |
|  |  | 6 | 5 | 4 |  |  |  |
|  |  | 3 | 2 | 1 |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |

|  |  |  |
| --- | --- | --- |
| 1 | 2 | 3 |
| 4 | 5 | 6 |
| 7 | 8 | 9 |

h[k,l]

k

l

* Impulzná odozva sa obracia podobne ako pri 1D
* Ľudovo povedané:
  + obrátime si impulznú odozvu, jazdím s ňou po obrázku, všetky bunky čo sú pod ňou vynásobím príslušným koeficientom v impulznej odozve, spočítam a nahodím do príslušného bodu vo výstupe
* Koeficienty impulznej odozvy voláme aj maska alebo konvolučné jadro

**Filter typu drôt**

* Impulzná odozva má 1 prvok o hodnote 1
* Nič sa nemení, obrázok sa len skopíruje

**Filter vyhladzovanie**

* 3x3 matica jedničiek
* Prídeme o časť kvality

**Sobelov filter - vertikálne hrany**

* Vyjdú nám aj záporné vzorky, musíme ich nejak previesť do [0..1] pre zobrazenie (napr. absolútna hodnota)

**Sobelov filter - vertikálne hrany**

**Odšumovanie**

* Napr. maska 9x9 , násobená 1/81
* Pozor ale na hrany, často sa rozmažú
* **Mediánový filter** - výber strednej hodnoty podľa veľkosti - trochu lepší výsledok

**Spektrálna analýza signálu**

* Úlohou je zistiť čo je v signáli na akej frekvencií
* Dôvod:
  + vizualizácia hodnôt
  + výpočet parametrov napr. na porovnanie tvárí
  + Filtrácia bez konvolúcie
  + kódovanie napr. JPEG
* Korelácia = určenie podobnosti:
* Analyzačný signál (obrázok 100x100)
  + horizontálna kosínusovka
  + nezáleží na k, preto sa mení len po riadkoch
  + vertikálna kosínusovka
  + nezáleží na l, preto sa mení len po stĺpcoch
  + mix
* zovšeobecnenie:
* rozumné rozsahy sú [0..1/2]
* skutočná frekvencia:
* problém znova s fázovým posunom - riešením je komplexná exponenciála
* vznikla nám 2D DFT - dvojdimenzionálna diskrétna Fourierova transformácia
* k,l - počítadlá pixelov na vstupe
* m,n - počítadlá frekvencií na výstupe
* m/K - normalizovaná vertikálna frekvencia
* n/L - normalizovaná horizontálna frekvencia
* takisto existuje spätná 2D DFT

**DCT - diskrétna cosinusová transformácia**

* nemáme radi 2D DFT kvôli komplexným číslam
* nemáme radi symetrie v obrázku
* 1D DCT
  + často sa normalizujú koeficienty
* 2D DCT
  + totálne rovnaké ale s dvoma rozmermi n1 a n2 v 2 sumách
  + rozmazaný obrázok, koeficienty obmedzené prahovaním, Lenna je horšia než predtým na DFT

**Sumár**

* Obraz je 2D signál
* Filtrácia - analógia s 1D FIR filtrami
* Jazdíme s maskou po obrázku, čo je pod ňou vynásobíme koeficientami a sčítame
* Frekvenčná analýza - rovnaká ako u 1D, kosínus nestačí
* 2D DFT - komplexné exponenciály
* Vo výslednej DFT matici je mnoho symetrií
* Dá sa v spektre filtrovať
* DCT
  + „2x” pomalšia báza
  + zložitejšia definícia
  + reálne koeficienty
  + žiadne symetrie ani komplexné čísla
  + využitie v JPEG