

Jasové a geometrické transformace

Václav Hlaváč

České vysoké učení technické v Praze

Český institut informatiky, robotiky a kybernetiky

160 00 Praha 6, Jugoslávských partyzánů 1580/3

<http://people.ciirc.cvut.cz/hlavac>, vaclav.hlavac@cvut.cz

také z Centra strojového vnímání, <http://cmp.felk.cvut.cz>

Osnova přednášky:

- ◆ Změna jasové stupnice.
- ◆ γ -korekce. Vyrovnání histogramu.
- ◆ Pseudobarva.
- ◆ Jasové korekce.
- ◆ Geometrické transformace.
- ◆ Interpolace hodnot obrazové funkce.

Předzpracování obrazu, úvod

Vstupem je obraz, výstupem je obraz.

Obraz se neinterpretuje.

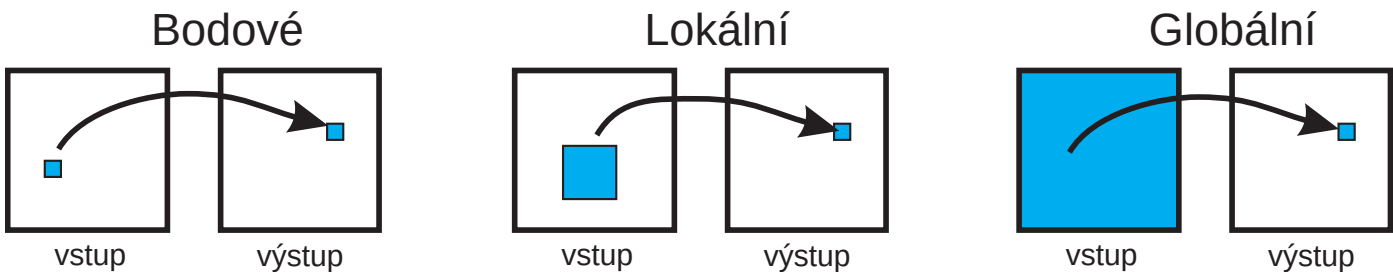
Cíl

- ◆ Zvýšení **kontrastu** (jen pro prohlížení obrazu člověkem).
- ◆ Potlačit **zkreslení** (např. korekce geometrického zkreslení díky zakřivenosti Země u družicového snímku).
- ◆ Odstranění **šumu**.
- ◆ **Zdůraznění charakteristik** obrazu pro další zpracování, např. nalézání hranových bodů.

Třídění metod předzpracování obrazu

Třídění **podle velikosti okolí právě zpracovávaného pixelu** (okamžitý pixel).

<i>Okolí</i>	<i>Příklad operace</i>	<i>Skutečně zpracovávané okolí</i>
Bodové	Transformace jasové stupnice	Stejná pro všechny pixely
	Jasové korekce	Jen okamžitý pixel
	Geometrické transformace	Teoreticky okamžitý pixel, prakticky malé okolí
Lokální	Lokální filtrace	Malé okolí
Globální	Fourierova transformace obrazu	Celý obraz
	Obnovení (restaurace) obrazu	

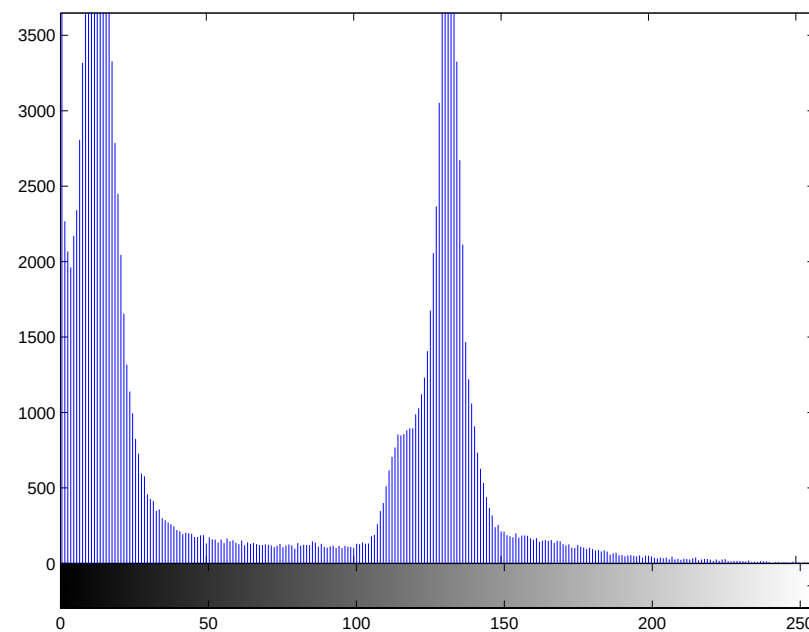


Histogram hodnot jasu

Histogram hodnot jasu je odhadem hustoty pravděpodobnosti jevu, že pixel bude mít určitou jasovou hodnotu.



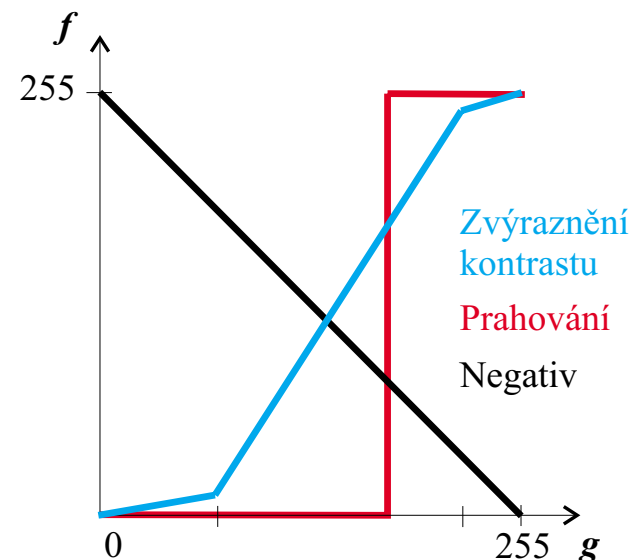
výchozí obraz



histogram hodnot jasu

Transformace jasové stupnice

- ♦ $g(i)$ – vstupní jasová stupnice,
- ♦ $f(i)$ – výstupní jasová stupnice,
- ♦ $i = 0, \dots, i_{\max}$ (maximální jas).

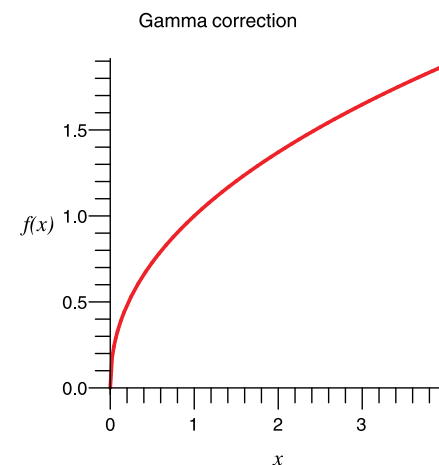
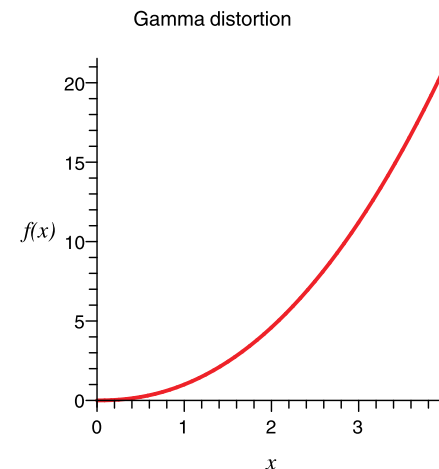


Poznámky:

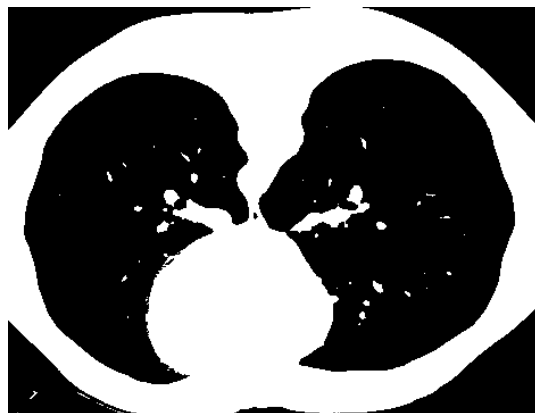
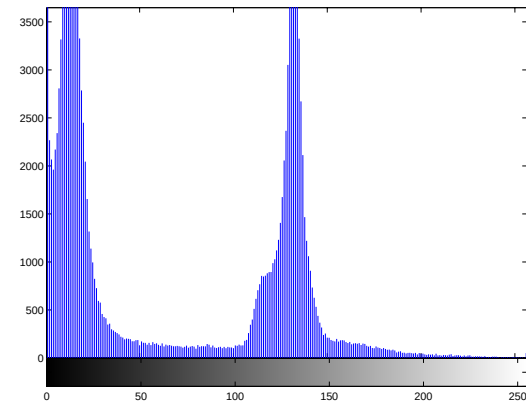
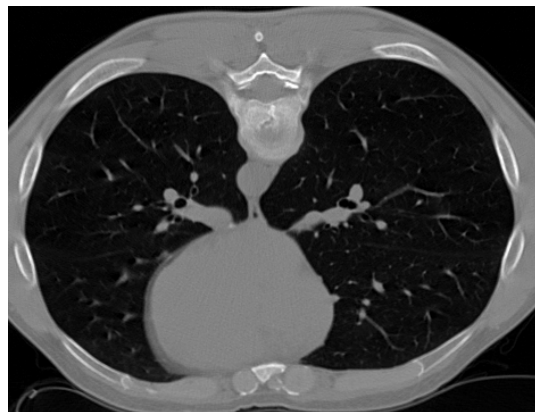
- ♦ HW realizace v zobrazovací kartě (např. ve VGA módu).
- ♦ Prahování převede šedotónový obraz na binární s černými a bílými oblastmi. Jak volit práh?

Nelinearita v jas, γ -korekce

- ◆ Proč kamery realizují nelineární transformaci jasů, tzv. γ -korekci?
- ◆ Ve vakuových obrazovkách televizí závisí jas na mřížkovém napětí U exponenciálně, tj. jas $\approx U^\gamma$, kde obvykle $\gamma = 2,2$.
- ◆ Aby se zachovala lineární přenosová funkce v celém řetězci, musí se tento jev v kamerách kompenzovat inverzní funkcí. Při vstupním označení E a výstupním napětí kamery U to odpovídá $U \approx E^{(1/\gamma)}$, obvykle $U = E^{0,45}$.
- ◆ I když u LCD monitoru jas závisí na vstupním napětí lineárně, používá se γ -korekce nadále kvůli zpětné kompatibilitě.



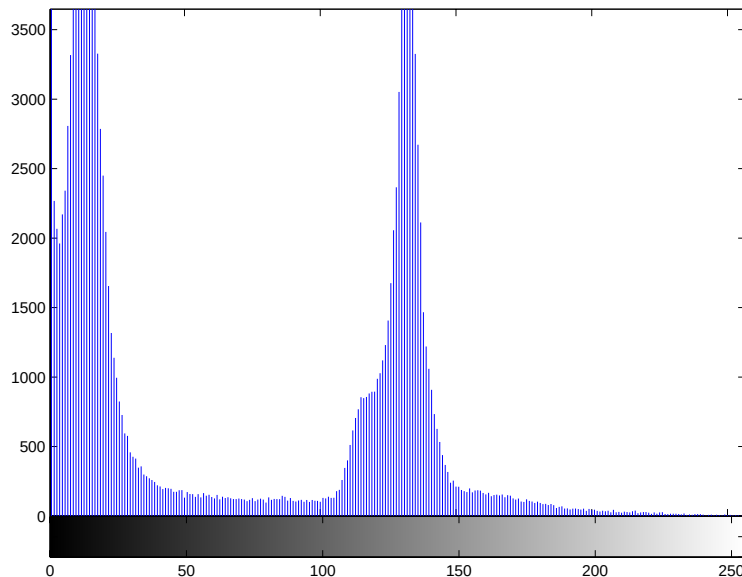
Příklad dvou různých prahů



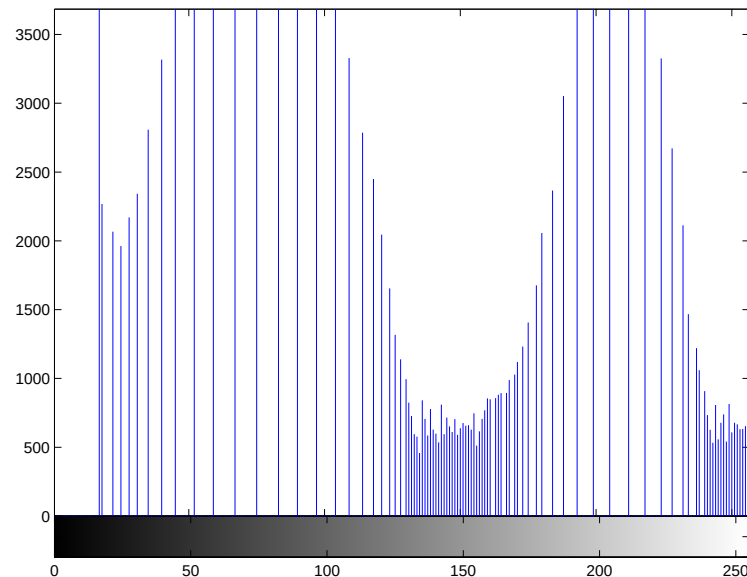
Ekvalizace histogramu

Cílem je:

- ♦ zvýšit kontrast úplným využitím jasové stupnice (pro pozorovatele – člověka),
- ♦ jasově obraz normalizovat (např. pro automatické porovnávání, vyhledávání).

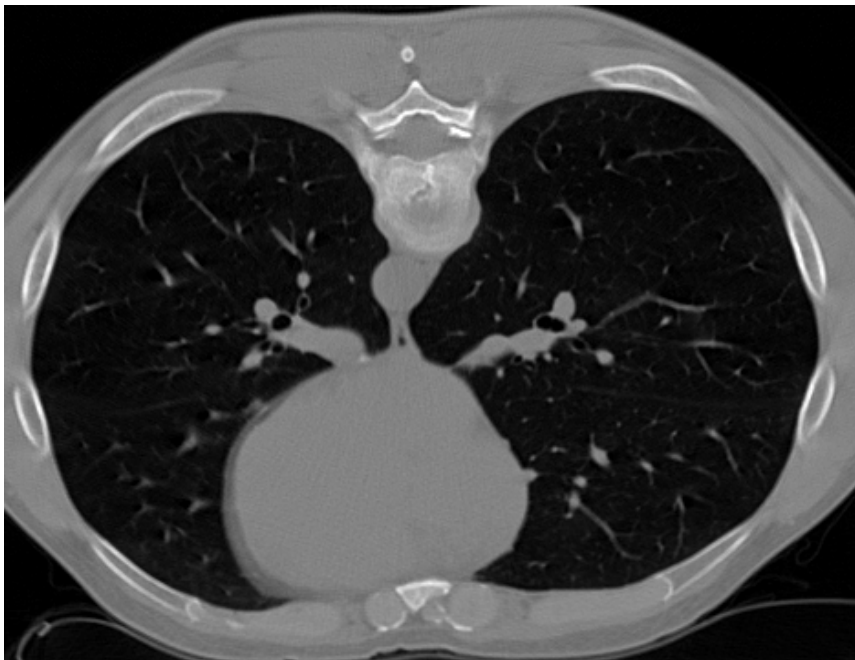


histogram původního obrazu

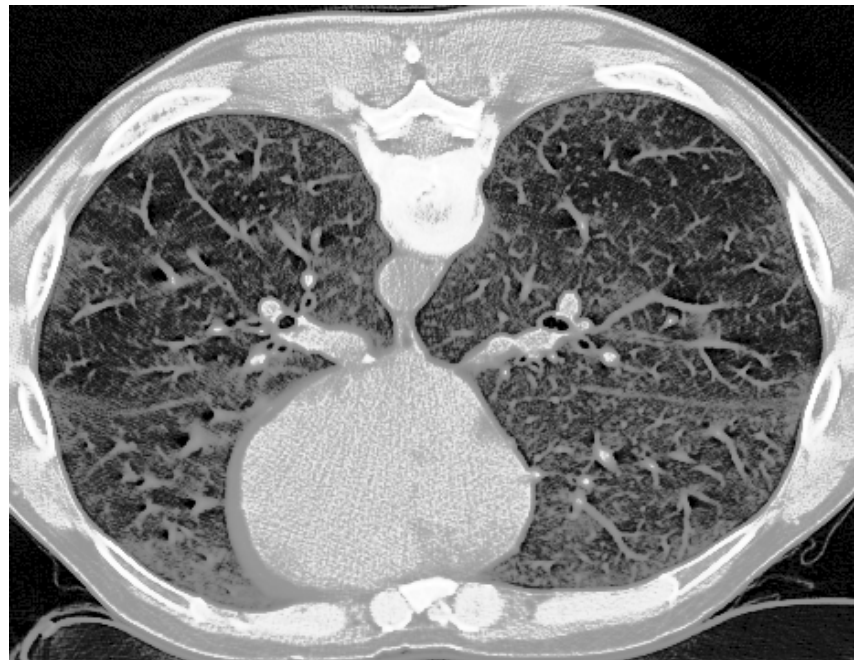


histogram po ekvalizaci

Zvýšení kontrastu po ekvalizaci histogramu



původní obraz



zvýšení kontrastu

Odvození ekvalizace histogramu

Vstup: histogram $H(p)$ vstupního obrazu s jasovou stupnicí $p = \langle p_0, p_k \rangle$.

Cíl: najít monotónní transformaci jasové stupnice $q = \mathcal{T}(p)$, aby výsledný histogram $G(q)$ byl rovnoměrný pro celý výstupní interval jasu $q = \langle q_0, q_k \rangle$.

$$\sum_{i=0}^k G(q_i) = \sum_{i=0}^k H(p_i) .$$

Ekvalizovaný histogram $\approx$ rovnoměrnému rozdělení

$$f = \frac{N^2}{q_k - q_0} .$$

Odvození ekvalizace histogramu (2)

Ideální rovnoměrný histogram získáme pouze pro spojitý případ.

$$\int_{q_0}^q G(s) \, ds = \int_{p_0}^p H(s) \, ds .$$

$$N^2 \int_{q_0}^q \frac{1}{q_k - q_0} \, ds = \int_{p_0}^p H(s) \, ds .$$

$$\frac{N^2(q - q_0)}{q_k - q_0} = \int_{p_0}^p H(s) \, ds .$$

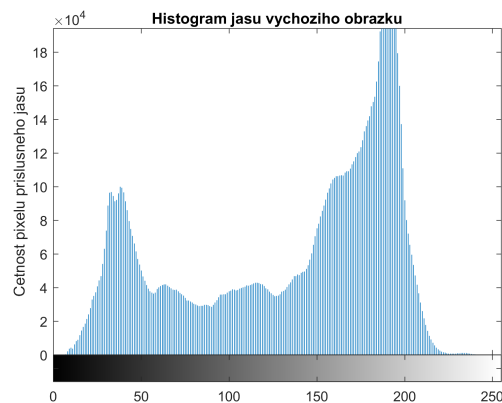
$$q = \mathcal{T}(p) = \frac{q_k - q_0}{N^2} \int_{p_0}^p H(s) \, ds + q_0 .$$

Diskrétní případ

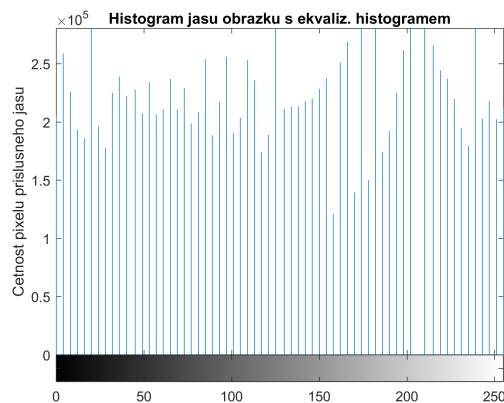
, kumulativní histogram

$$q = \mathcal{T}(p) = \frac{q_k - q_0}{N^2} \sum_{i=p_0}^p H(i) + q_0 .$$

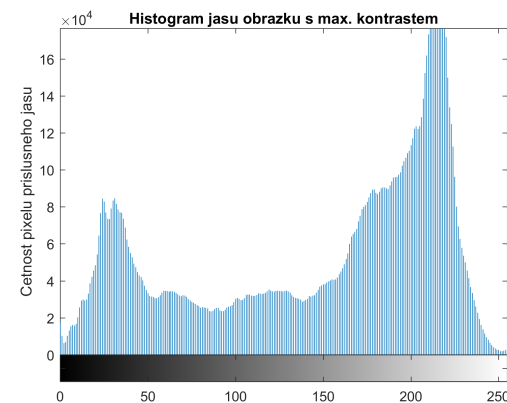
Příklad zvýšení kontrastu ekvalizací histogramu



výchozí obrázek



ekvalizovaný histogram

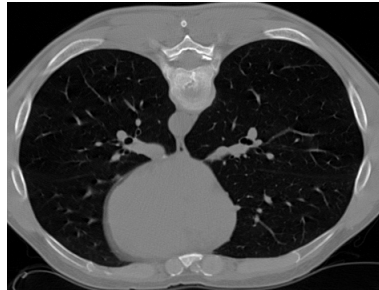


maximální kontrast (Photoshop)

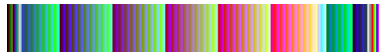
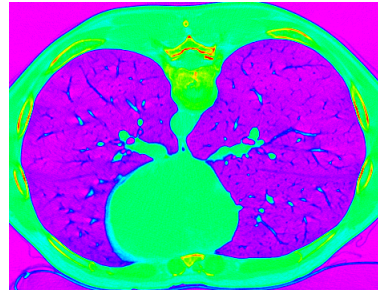
Pseudobarva



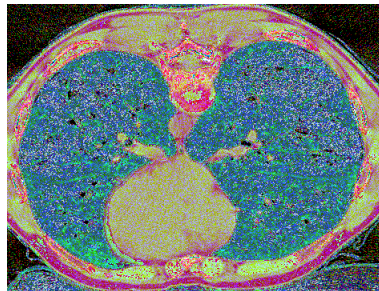
Beze změny



Spektrum



System (Windows)



Černé těleso



- ◆ Pseudobarva zobrazuje šedotónový obrázek jako barevný obrázek tak, že hodnotu intenzity převádí na barvu podle vyhledávací tabulky nebo funkce.
- ◆ Funkci realizující zobrazení se říká paleta.
- ◆ Pseudobarva se obvykle používá pro zobrazení jednokanálových dat (např. teploty, hloubky).

Jasové korekce

- ◆ Nový jas $f(i, j)$ závisí na poloze i, j vstupního obrazu $g(i, j)$.
- ◆ Často multiplikativní model poruchy: $f(i, j) = e(i, j) g(i, j)$.

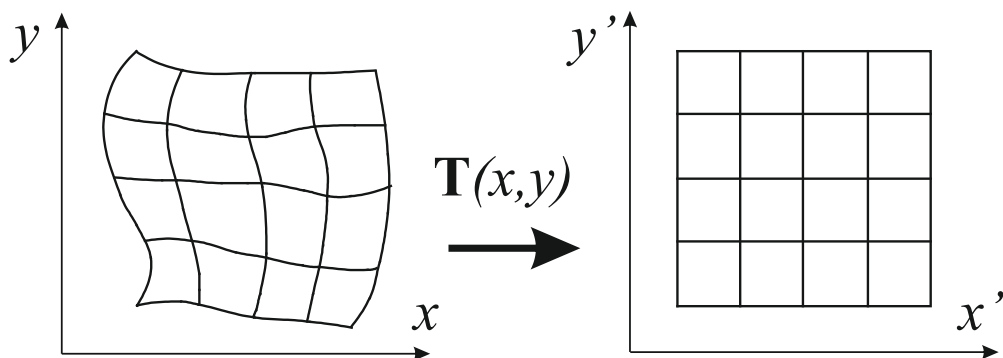
2 postupy:

1. Opravné koeficienty získány **snímáním etalonové plochy** známého jasu c , např. při kompenzaci nerovnoměrného osvětlení (vypnout AGC!). Po sejmutí získáme

$$f_c(i, j) = e(i, j) c \quad \Rightarrow \quad e(i, j) = \frac{f_c(i, j)}{c}$$

2. **Proložení pozadí analytickou plochou** a její odečtení od původního obrazu.

Geometrická transformace, motivace ve 2D



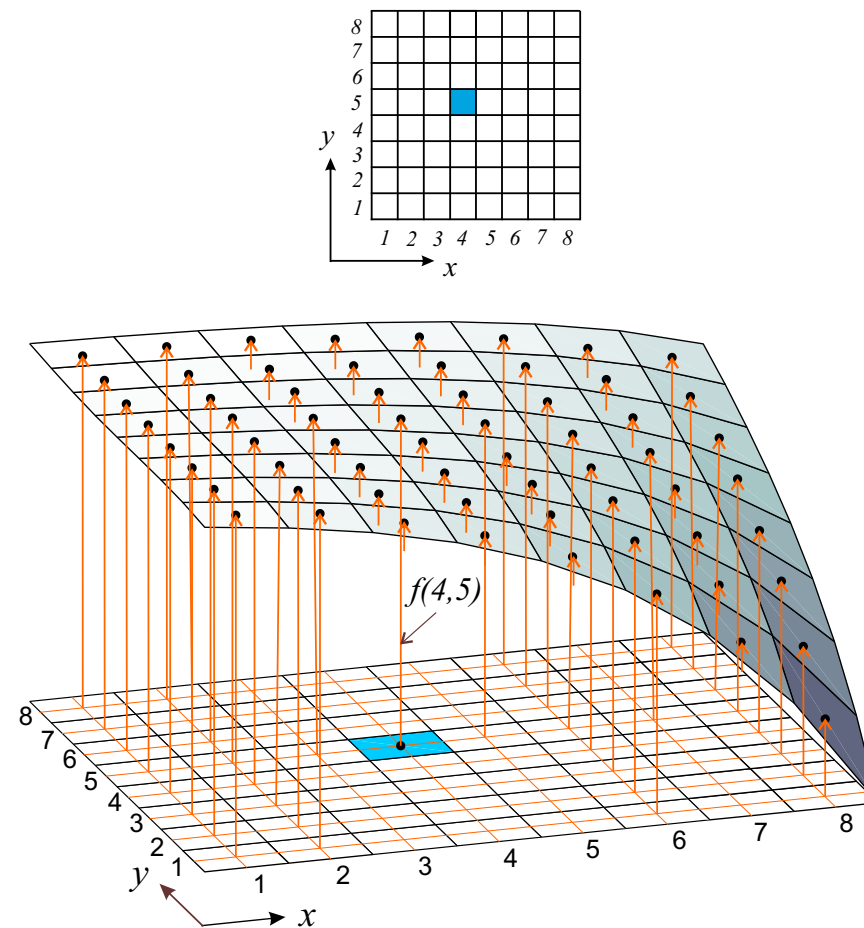
Rozepíšeme vektorovou transformaci $\mathbf{T}$ do dvou složek

$$x' = T_x(x, y), \quad y' = T_y(x, y).$$

- ◆ Co musí být splněno, aby vzájemný převod byl jednoznačný?
- ◆ Geometrické transformace se **se používají pro**
 - zvětšování, posouvání, pootáčení, zkosení 2D obrazu (tj. podle předem známé transformace).
 - odstraňování geometrických zkreslení (příslušná geometrická transformace se často odhaduje z obrazu).

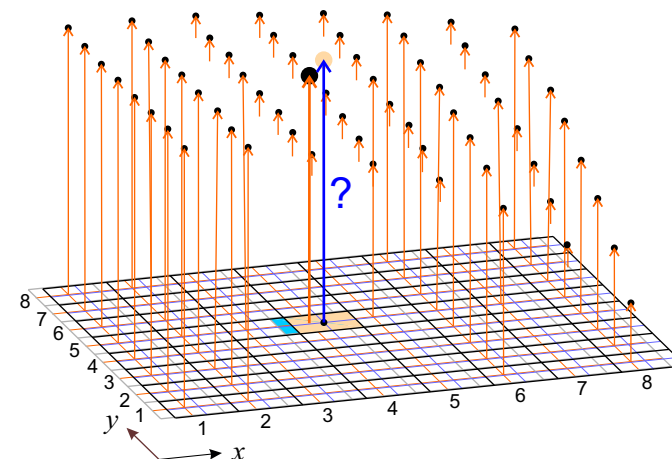
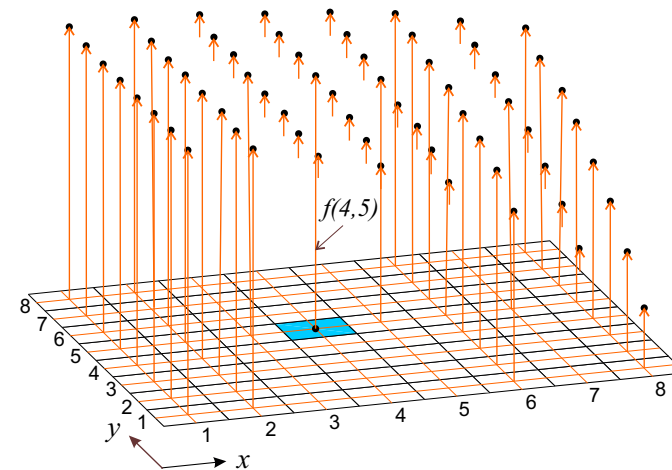
Topografický pohled na digitální intenzitní obraz

- ◆ Vzpomeňme, že jsme si v úvodní přednášce představovali obrázek $f(x, y)$ jako krajinu.
- ◆ Nechť hodnoty jasu $f(x, y)$ odpovídají výšce v krajině v místě daném souřadnicemi x, y .
- ◆ Digitální obraz poskytuje pouze vzorky v diskrétní mřížce ilustrované na obrázku oranžovými šipkami.



Topografický pohled na geom. transform. obraz

- ◆ Digitální obraz poskytuje pouze vzorky v diskrétním rastru. Viz obrázek vpravo nahoře.
- ◆ Uvažujme **příklad**: Obrázek jsem posunuli o polovinu pixelu ve směru osy x . Viz obrázek vpravo dole.
- ◆ **Problém**: Nemáme k dispozici hodnotu obrazové funkce $f(x + 0.5, y)$.
- ◆ **Řešení**: Interpolujeme chybějící hodnotu ze sousedních hodnot obrazové funkce.
Proč pouze ze sousedních hodnot?



Nutné dva kroky transformace kvůli diskrétnímu rastru

1. *Transformace souřadnic bodu* počítá se nová poloha každého bodu ve spojitých souřadnicích (reálná čísla), protože výsledek může být mimo rastr.

Problémy:

- ◆ Část obrazu leží mimo obraz.
 - ◆ Transformace díky nutné aproximaci nejsou invertovatelné.
2. *Aproximace jasové funkce* hledá celočíselnou hodnotu jasu v celočíselné pozici, která nejlépe odpovídá nově vypočítané neceločíselné poloze x', y' .

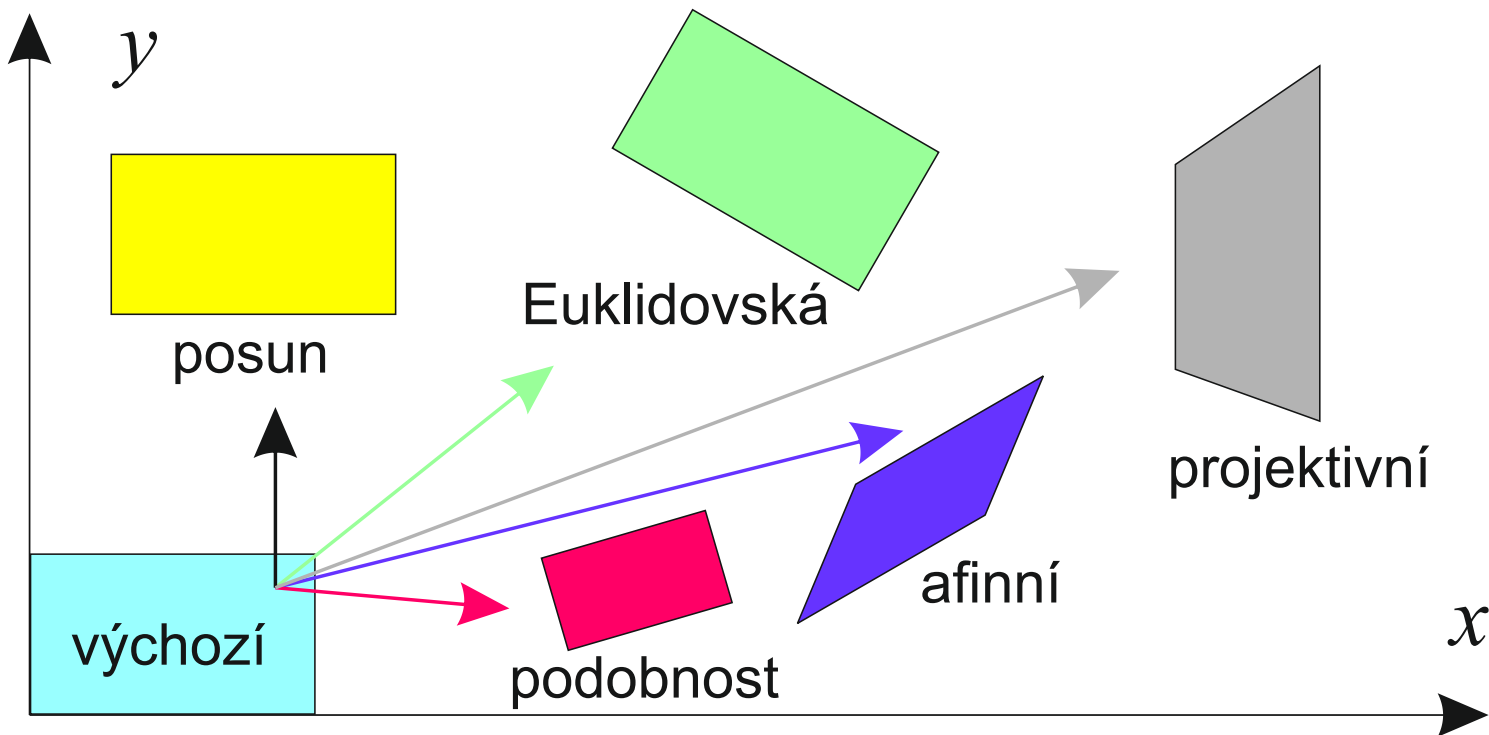
Transformace souřadnic bodu

- ◆ Geometrická transformace $x' = T_x(x, y)$, $y' = T_y(x, y)$ se obvykle **aproximuje polynomem** m -tého stupně.

$$x' = \sum_{r=0}^m \sum_{k=0}^{m-r} a_{rk} x^r y^k, \quad y' = \sum_{r=0}^m \sum_{k=0}^{m-r} b_{rk} x^r y^k.$$

- ◆ Vztah je lineární vzhledem ke koeficientům a_{rk} , b_{rk} .
- ◆ Neznámé koeficienty se získají řešením **soustavy lineárních rovnic**, v nichž slouží jako známé hodnoty **dvojice sobě odpovídajících (vlíčovacích) bodů** (x, y) and (x', y') .
- ◆ Kdyby se soustava lineárních rovnic řešila deterministicky, stačilo by jen několik párů vlíčovacích bodů. Počet rovnic se snadno určí z počtu neznámých koeficientů v soustavě rovnic.
- ◆ V praxi se ale používá **mnoho párů vlíčovacích bodů**, což vede na **přeurčenou soustavu lineárních rovnic**. Neznámé koeficienty a_{rk} , b_{rk} se odhadují **metodou nejmenších čtverců**.

Základní geometrické transformace v rovině



Inspirace R. Szeliski.

Bilineární, afinní transformace souřadnic

Když se geometrická transformace v závislosti na pozici v obraze příliš náhle nemění, postačují aproximační polynomy nízkého stupně $m = 2$ nebo $m = 3$.

◆ Bilineární transformace

$$\begin{aligned}x' &= a_0 + a_1x + a_2y + a_3xy, \\y' &= b_0 + b_1x + b_2y + b_3xy.\end{aligned}$$

◆ Afinní transformace, která je speciálnější, zahrnuje v praxi potřebnou rotaci, translaci a zkosení.

$$\begin{aligned}x' &= a_0 + a_1x + a_2y, \\y' &= b_0 + b_1x + b_2y.\end{aligned}$$

Homogenní souřadnice $\Rightarrow$ zápis maticemi

- ◆ Homogenní souřadnice jsou obvyklé v teoretické mechanice, projektivní geometrii, počítačové grafice a robotice.
- ◆ Základní myšlenkou je reprezentovat bod ve vektorovém prostoru o jednu dimenzi větším.
- ◆ Bod $[x, y]^T$ se v homogenních souřadnicích vyjádří ve 3D vektorovém prostoru jako $[\lambda x, \lambda y, \lambda]^T$, kde $\lambda \neq 0$.
- ◆ Pro jednoduchost se obvykle používá jedno z nekonečně mnoha vyjádření $[x, y, 1]^T$.

Afinní transformace maticově

Afinní zobrazení se po zavedení homogenních souřadnic vyjádří

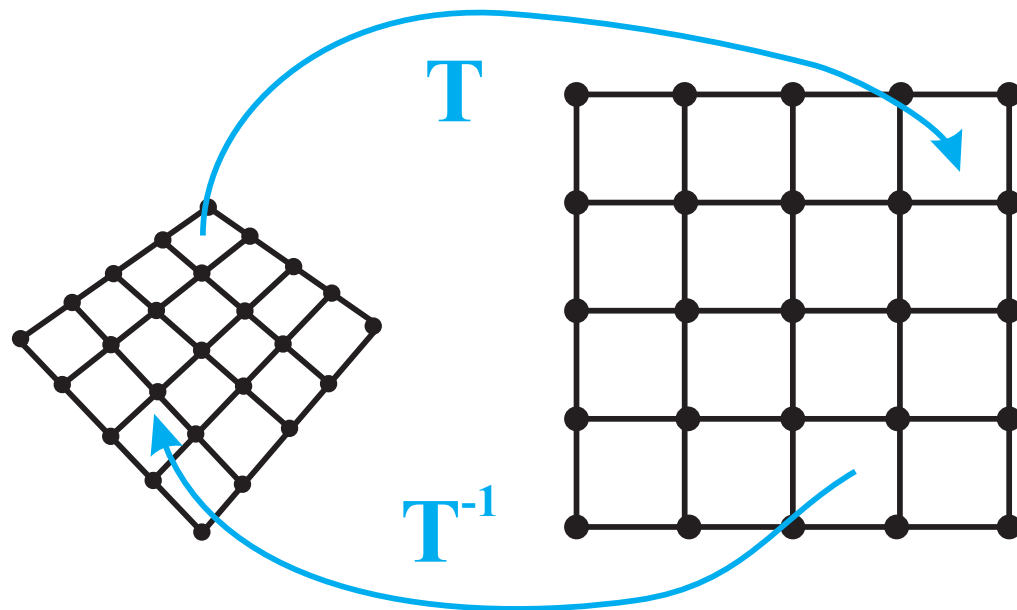
$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} a_1 & a_2 & a_0 \\ b_1 & b_2 & b_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} .$$

Poznámky:

- ◆ Souvislost s jazykem PostScript (v prodeji od 1984; Apple LaserWriter 1985).
- ◆ Složitější geometrické transformace (zkreslení) se aproximují tak, že se obraz rozdělí na menší obdélníkové podobrazy. Pro každý z podobrazů se použije jednodušší geometrická transformace (např. afinní) odhadnutá z vlíčovacích bodů.

Použití geometrické transformace duální vyjádření

- ◆ Vstupní obraz se zobrazí pomocí transformace $\mathbf{T}$ do výstupního obrazu.
- ◆ Duální vyjádření:
 - Dopředná transformace:
 $(x', y') = \mathbf{T}(x, y)$.
 - Zpětná transformace:
 $(x, y) = \mathbf{T}^{-1}(x', y')$.
- ◆ Obě transformace se liší kvůli nezbytnosti jasové aproximace v diskrétní mřížce.



Přímá a zpětná transformace, srovnání

Přímá transformace

- ◆ Souřadnice ve výstupním obraze (x', y') mohou ležet mimo rastr.
- ◆ Dva a více vstupních pixelů se může zobrazit do stejného výstupního pixelu.
- ◆ Některým výstupním pixelům nemusí být přiřazena hodnota. Objeví se díry.

Zpětná transformace

- ◆ Pro každou polohu pixelu ve výstupním obraze (x', y') se hledá poloha ve vstupním obraze (x, y) pomocí $\mathbf{T}^{-1}$, v jejímž okolí se bude aproximovat jasová hodnota z dostupných diskrétních vzorků.
- ◆ Díry nevznikají.
- ◆ Potíž: $\mathbf{T}^{-1}$ vždy nemusí existovat.

Aproximace obrazové funkce

- ◆ Principiální správnou odpověď poskytuje [teorie aproximace](#). Ze vzorku odhadneme spojitou 2D funkci.
- ◆ Odhadovaná 2D funkce se lokálně aproximuje polynomem. (Proč lokálně?)
- ◆ Interpolace je zvláštní případ aproximace, kdy odhadovaná hodnota leží “mezi” známými vzorky.

Q : Proč je aproximace polynomy oblíbená?

A₁: Polynom má několik málo parametrů. Nejlepší hodnoty těchto parametrů se hledají optimalizační metodou. Polynomy mají spojité derivace, a tak lze buď optimum odvodit analyticky nebo využít numerickou optimalizaci gradientní metodou.

A₂: Když se parametry aproximačního polynomu odhadují experimentálně, je obvykle mnohem více příkladů než parametrů. Optimální parametry se odhadují pomocí metody nejmenších čtverců. S ohledem na A₁, lze potřebné derivace spočítat analyticky.

Tři používané interpolační metody

Pro interpolaci hodnot 2D obrazové funkce, např. po geometrických transformacích, se používají tři aproximační metody:

1. **Nejbližší soused.**
2. **Bilineární interpolace.** Lineární 1D interpolace se použije nejdříve v jedné a potom druhé dimenzi.
3. **Bikubická interpolace.**

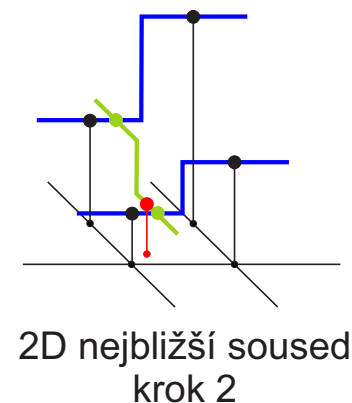
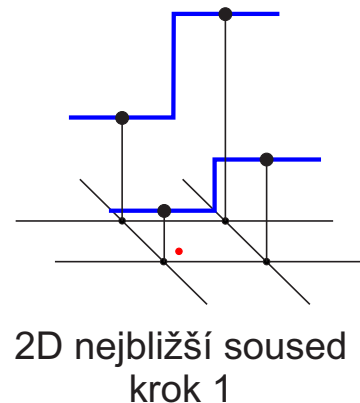
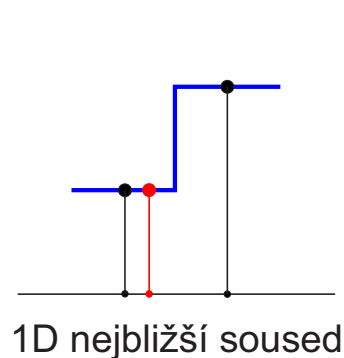
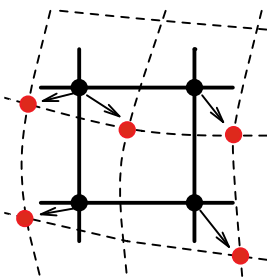
Pozn.: “Bi” na začátku názvu bilineární/bikubická znamená, že 1D interpolace se použije nejdříve v jedné a potom druhé dimenzi.

Pro každou ze tří metod ukážeme myšlenku interpolační funkce a na třech obrázcích ilustrujeme postup. Nejdříve ukážeme, jak by interpolace vypadala v 1D případě. Následovat budou dva dílčí kroky 1D interpolace ve 2D případě.

Až se seznámíme se všemi třemi metodami, srovnáme na jedné průsvitce chování těchto metod na praktickém příkladu, a to zvýšení rozlišení textu 20 krát.

Interpolace metodou nejbližšího souseda

- ◆ Červený bod (x, y) uprostřed černými čarami ohraničeného čtverce označuje polohu bodu po geometrické transformaci. K němu se hledá hodnota obrazové funkce $f(x, y)$. Čárkované čáry označují hypotetický vzorkovací rastr po geometrické transformaci.
- ◆ Interpolace metodou nejbližšího souseda přiřadí hodnotě obrazové funkce $f(x, y)$ hodnotu nejbližšího (černého) bodu původního vzorkovacího rastru, tj. odpovídající hodnotě ve vzorkovaném obraze g_s , $f(x, y) = g_s(\text{round}(x), \text{round}(y))$.

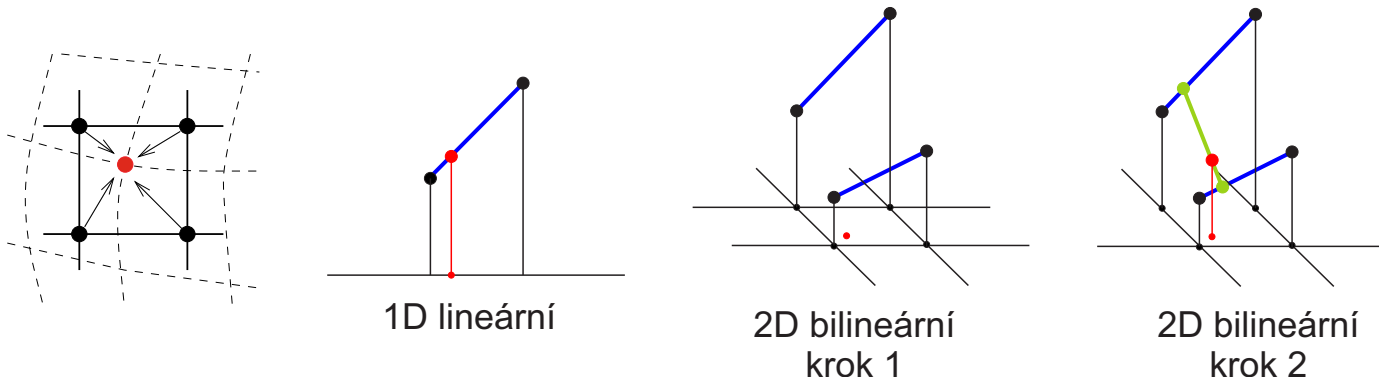


Bilineární interpolace

Uvažujme jeden (červený) bod (x, y) v diskrétním rastru po geometrické transformaci ležící mimo známé vzorky v původním rastru, obrázek vlevo. Známe čtyři hodnoty obrazové funkce f v původním rastru (černé body). Aproximujeme hodnotu spojitě obrazové funkce $f(x, y)$ pro červený bod (x, y) .

V **prvním kroku** se v jednom souřadném směru rastru 1D lineárně proloží dvě dvojice krajních hodnot obrazové funkce dvěma modrými úsečkami. Druhý obrázek zleva ukazuje situaci v 1D, třetí obrázek zleva ve 2D.

Ve **druhém kroku**, obrázek vpravo, se najdou na těchto (modrých) krajních proloženíh místa (zelené body) odpovídající bodu (x, y) . Tato místa se lineárně proloží (zelená úsečka). Získá se hledaná hodnota $f(x, y)$ (červený bod).



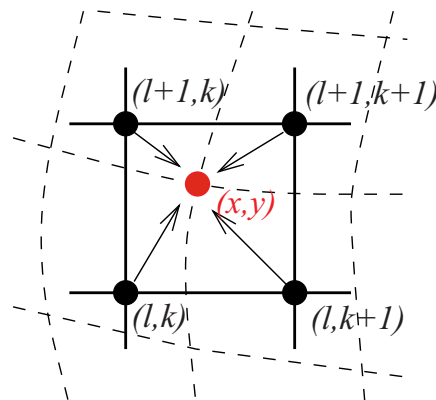
Bilineární interpolace, výpočet hodnoty

Interpolovaná hodnota $f(x, y)$ se vypočte jako lineární kombinace čtyř vzorků ze vzorkovací mřížky. Vliv hodnot každého ze čtyř bodů mřížky v lineární kombinaci je úměrný jeho blízkosti k poloze interpolovaného bodu.

$$\begin{aligned}
 f(x, y) &= (1 - a)(1 - b) g_s(l, k) + a(1 - b) g_s(l + 1, k) \\
 &+ b(1 - a) g_s(l, k + 1) + ab g_s(l + 1, k + 1),
 \end{aligned}$$

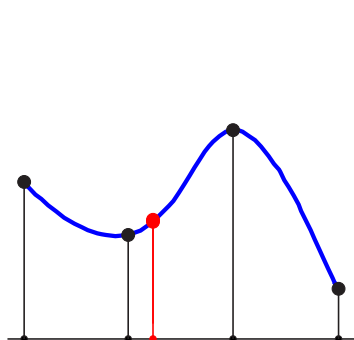
kde

$$\begin{aligned}
 l &= \text{ceil}(x), \quad a = x - l, \\
 k &= \text{ceil}(y), \quad b = y - k.
 \end{aligned}$$

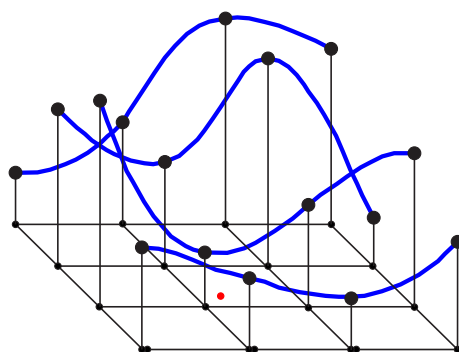


Bikubická interpolace

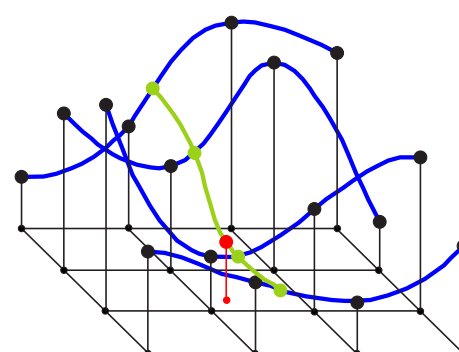
- ♦ Vstupem je 16 vzorků f_s ve vzorkovací mřížce.
- ♦ Výstupem je interpolovaná hodnota obrazové funkce $f(x, y)$ pro bod (x, y) (červený bod) ležící mimo vzorkovací mřížku.
- ♦ V **prvním kroku** se interpoluje v jednom směru souřadnic čtyřmi 1D kubickými polynomy (čtyři modré křivky). Ve **druhém kroku** se najdou čtyři body (zelené body) odpovídající poloze (x, y) . Tyto body se proloží jedním 1D kubickým polynomem a najde se hledaná hodnota (červený bod).



1D kubická



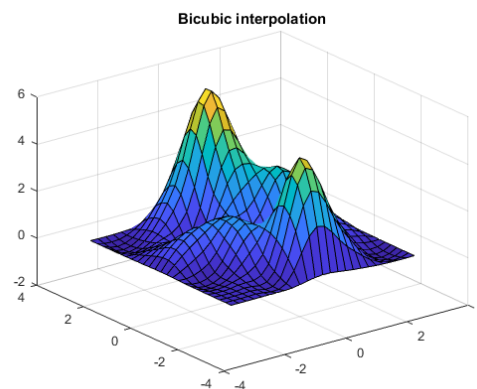
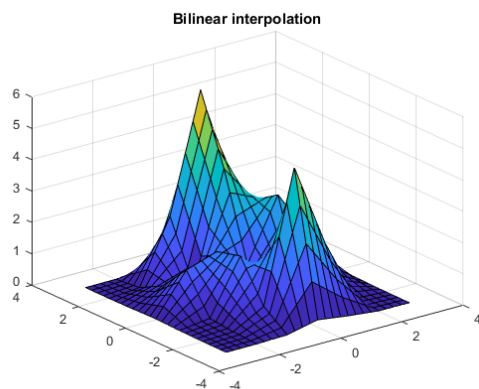
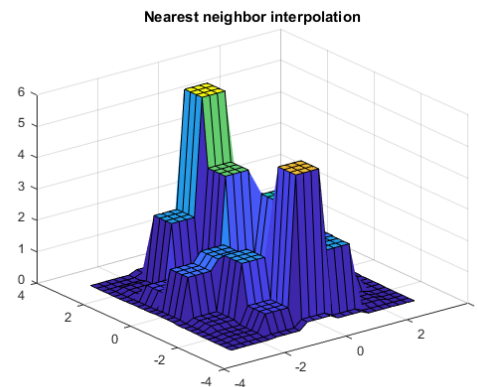
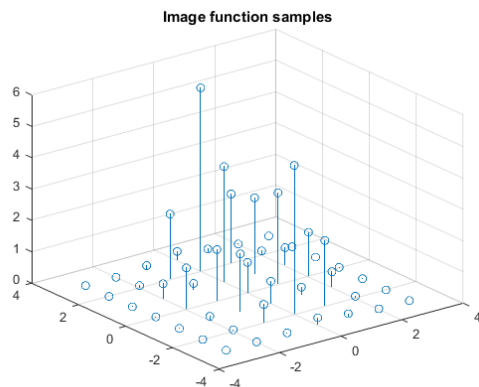
2D bikubická
krok 1



2D bikubická
krok 2

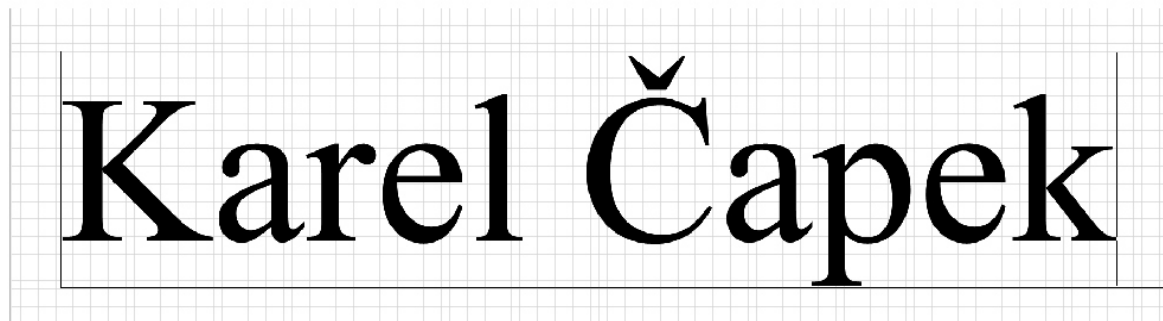
Tři interpolační metody ilustrované jako povrchy

Vstupní data jsou záměrně v malém rozlišení 7×7 .



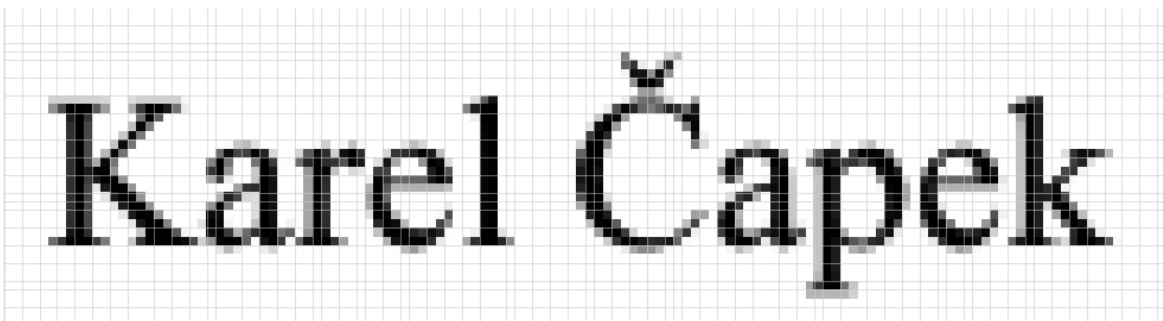
Výchozí obrázek pro srovnání tří interpolací

Times Roman
Ideal
36 x 132 px



Karel Čapek

Times Roman
Real
36 x 132 px



Karel Čapek

Zvětšování obrazu 20x, srovnání tří metod



34/36

Ideal
Times Roman

Karel Čapek

Nearest neighbor

Karel Čapek

Bilinear

Karel Čapek

Bicubic

Karel Čapek

Aproximace jako 2D konvoluce

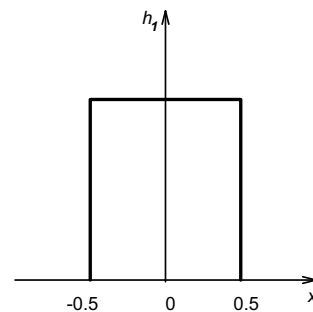
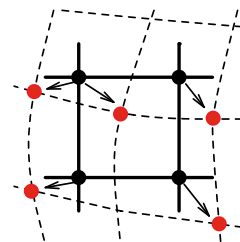
- ◆ Místo původní spojité obrazové funkce $f(x, y)$ známe její vzorkovanou verzi $f_s(l, k)$.
- ◆ Výsledkem aproximace (interpolace) je jas $f_n(x, y)$, kde index n rozlišuje jednotlivé interpolační metody. Jas lze vyjádřit jako dvojrozměrnou konvoluci

$$f_n(x, y) = \sum_{l=-\infty}^{\infty} \sum_{k=-\infty}^{\infty} f_s(l, k) h_n(x - l, y - k) .$$

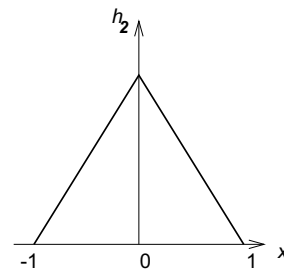
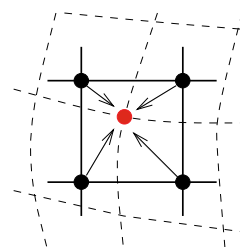
- ◆ Funkce h_n je **interpolační jádro**.
- ◆ Obvykle se používá interpolační jádro pokrývající jen malé okolí zpracovávaného bodu, aby se ušetřily výpočty. Vně tohoto okolí je hodnota interpolačního jádra h_n nulová.

Tři konvoluční jádra, ilustrace obrázky v 1D

Nearest neighbor



Bilinear interpolation



Bicubic interpolation

