

→ biorąc pod uwagę tylko najbliższy piksel,  $O(1nm)$

Nearest Neighbour interpolation → najbliższe, ale efektem jest zazwyczaj zdjęcie niesymetryczne rozkładanej → cv2 "nearest"

Przykład - ze zdjęcia  $2 \times 2$  na zdjęcie  $4 \times 4$  - system pozygonowania jest od  $(1,1)$  - lewy górný róg

$$\begin{array}{|c|c|} \hline P_{11} & P_{12} \\ \hline P_{21} & P_{22} \\ \hline \end{array} \text{ interpoluje do} \quad \begin{array}{|c|c|} \hline Q_{11} & Q_{12} \\ \hline Q_{21} & Q_{22} \\ \hline \end{array}$$

$P_{11}, P_{12}, P_{21}, P_{22}$

zatem stała transformacji  $s = \frac{\text{wysokość obrazu}}{\text{wysokość oryginału}} = \frac{2}{1} = 2$

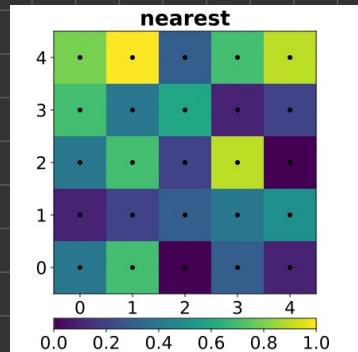
← na własne wartości interpolowane

Dla  $Q_{11}$ :  $(1,1) \cdot s = (\frac{1}{2}, \frac{1}{2}) \rightarrow$  najbliższy (euclidean)  $P_{11}$ , stąd  $Q_{11} = P_{11}$

$Q_{12}$ :  $(1,2) \cdot s = (\frac{1}{2}, 1) \rightarrow$  najbliższy  $P_{11} \Rightarrow Q_{12} = P_{11}$

$Q_{23}$ :  $(2,3) \cdot s = (\frac{3}{2}, \frac{3}{2}) \rightarrow$  najbliższy  $P_{22} \Rightarrow Q_{23} = P_{22}$

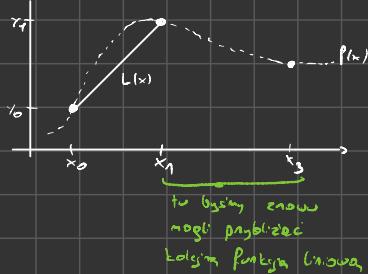
Dla macierzy niezależnych menią oddzielnie  $s_x, s_y$



Interpolacja liniowa → przybliżanie wartości funkcji ciągłej punktami liniowymi nie predstawiło oznaczonych wartościach  $f$  w konicach

Niech  $f: \langle x_0, x_1 \rangle \times \mathbb{R} \rightarrow \mathbb{R}$ ,  $f \in C(\langle x_0, x_1 \rangle)$ ,  $f(x_i) = y_i$  dla  $i \in \{0, 1\}$ . Liniowa interpolacja  $f$  nie predstawiła

$\langle x_0, x_1 \rangle$  rezygnowała funkcja  $L(x) = y_0 + \frac{y_1 - y_0}{x_1 - x_0} (x - x_0)$



→ biorąc pod uwagę 6 najbliższych pikseli (kwadrat  $3 \times 3$ ),  $O(6nm)$

Interpolacja biliniowa (dualiniowa) → złożenie 2 interpolacji liniowych → cv2 "bilinear"

Rozszerzenie interpolacji liniowej na funkcje 2 zmiennej - niech  $f: \langle x_0, x_1 \rangle \times \langle y_0, y_1 \rangle \subset \mathbb{R}^2 \rightarrow \mathbb{R}$ ,  $f \in C(\langle x_0, x_1 \rangle \times \langle y_0, y_1 \rangle)$

Niech  $Q_{00} = (x_0, y_0)$ ,  $Q_{10} = (x_1, y_0)$ ,  $Q_{01} = (x_0, y_1)$ ,  $Q_{11} = (x_1, y_1)$

Przeprowadzamy kolejną interpolację liniową względem  $\langle x \rangle$ :

$$f(R_0) \approx \frac{x_1 - x}{x_1 - x_0} f(Q_{00}) + \frac{x - x_0}{x_1 - x_0} f(Q_{10}) \quad \text{gdzie } R_0 = (x, y_0) \quad \text{dla } x \in \langle x_0, x_1 \rangle$$

$$f(R_1) \approx \frac{x_1 - x}{x_1 - x_0} f(Q_{01}) + \frac{x - x_0}{x_1 - x_0} f(Q_{11}) \quad \text{gdzie } R_1 = (x, y_1) \quad \text{dla } x \in \langle x_0, x_1 \rangle$$

Następnie przeprowadzamy interpolację względem  $\langle y \rangle$ :

$$f(P) \approx \frac{y_1 - y}{y_1 - y_0} f(R_0) + \frac{y - y_0}{y_1 - y_0} f(R_1) \quad \text{gdzie } P = (x, y) \in \langle x_0, x_1 \rangle \times \langle y_0, y_1 \rangle$$

Jeli przyjmujemy system współrzędnych, w których znane wartości  $f$  znajdują się w punktach

$Q_{00}(0,0)$ ,  $Q_{10}(1,0)$ ,  $Q_{01}(0,1)$ ,  $Q_{11}(1,1)$  to wówczas uproszcza się do

$$f(x,y) \approx |1-x| \cdot |1-y| \cdot f(0,0) + |x| \cdot |1-y| \cdot f(1,0) + |1-x| \cdot |y| \cdot f(0,1) + |x| \cdot |y| \cdot f(1,1) \quad \text{dla } (x,y) \in \langle x_0, x_1 \rangle \times \langle y_0, y_1 \rangle$$

Bez takiego systemu dostajemy

$$f(x,y) \approx \frac{1}{(x_1 - x_0)(y_1 - y_0)} \cdot |x_2 - x| \cdot |x - x_1| \cdot |y_2 - y| \cdot |y - y_1| \cdot |P(Q_{00}) - P(Q_{01})| \cdot |P(Q_{00}) - P(Q_{10})| \cdot |P(Q_{01}) - P(Q_{10})| \cdot |P(Q_{00}) - P(Q_{11})| \cdot |P(Q_{01}) - P(Q_{11})| \cdot |P(Q_{10}) - P(Q_{11})|$$

Głosunkowo szybka, może powodować rozmycie szczegółów

Jeli ogólnie to cv2 - przykład: zdjęcie  $2 \times 2 \rightarrow 4 \times 4$  (stała  $s = 2$ )

Niech zatem  $\begin{array}{|c|c|} \hline P_{11} & P_{12} \\ \hline P_{21} & P_{22} \\ \hline \end{array}$ ,  $\begin{array}{|c|c|} \hline Q_{11} & Q_{12} \\ \hline Q_{21} & Q_{22} \\ \hline \end{array}$ . Mamy  $Q_{11} = P_{11}$ ,

$$\begin{array}{c} 1 \\ P_{11} \\ P_{12} \\ P_{21} \\ P_{22} \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \\ 7 \\ 8 \\ 9 \\ 10 \\ 11 \\ 12 \\ 13 \\ 14 \\ 15 \\ 16 \end{array}$$

$$\begin{aligned} Q_{12} &= 0.75 P_{11} + 0.25 P_{12} \\ Q_{21} &= 0.25 P_{11} + 0.75 P_{12} \\ Q_{14} &= P_{12} \\ Q_{22} &= \frac{0.75}{2} P_{11} + \frac{0.25}{2} P_{12} + \frac{0.75}{2} P_{21} \end{aligned}$$

Piksele w rogach obrazu są takie same jak w oryginalnym piksele (poza rogami) na krawędziach obrazu sumują się

→ biorąc pod uwagę najbliższe 16 pikseli (kwadrat  $4 \times 4$ ), cv2 = "bicubic",  $O(16nm)$

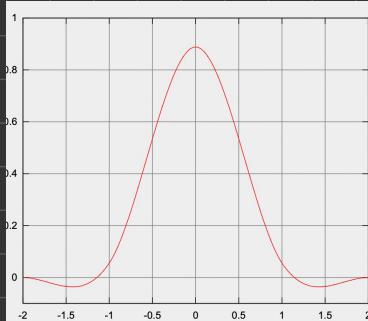
Bicubic interpolation

Niech  $f: I \subset \mathbb{R}^2 \rightarrow \mathbb{R}$ ,  $f \in C(I)$ ,  $I$  - domeny,  $P_{ij} = (i,j) \in I$  dla  $i, j \in \{0, 1\}$  - dla wartości innych przed obliczeniem

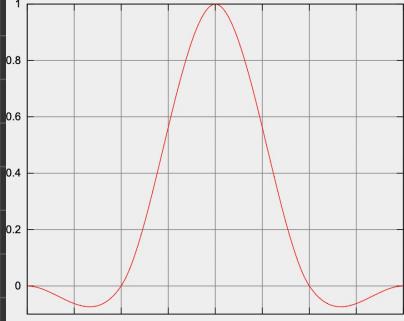
$$\text{znamy } \frac{\partial f}{\partial x} f(P_{ij}), \frac{\partial f}{\partial y} f(P_{ij}), \frac{\partial^2 f}{\partial x^2} f(P_{ij})$$

jeżeli jesteśmy tak aby spełniały

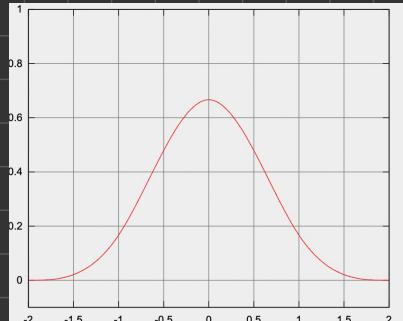




Mitchell-Netravali



## Catmull-Rom



### Cubic B-Spline

Area interpolation (metode resampling) → vzdáleností záleží od klasického pixelního broušení pod úhlem - od  $O(n^2)$  do  $O(n^2 \cdot m^2)$

Obrona średnia, wartości pikseli na określonym obszarze. Dobra jedynie do pomniejszania.

Czesem ulega się rozwinięciu warstw metygmatycznych / mineralnych → norzej szybkie, gładsza jakość i rozpiętościowe

Opoortu na funkcji  $\text{sinc}(x) = \frac{\sin(\pi x)}{\pi x}$  dla  $x \neq 0$  i 1 dla  $x=0$ . To ma jednak nieskończone wspołr.  $\Rightarrow$  musieliśmy uwzględnić wszystkie punkty obrazu  $\Rightarrow$  jest ograniczona przez funkcję Lamegoa  $L(x) = \int_0^x \text{sinc}(t) \text{sinc}\left(\frac{x}{t}\right) dt$

gärt & okresla nominer done - die Lanzos-3  $\alpha=3$ , die Lanzos-5  $\alpha=5$ .

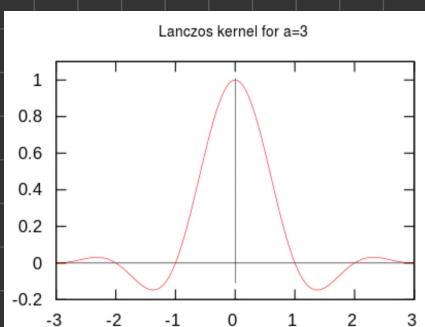
Die 2-wertigen logischen Funktionen  $L(x, y) = L(x)L(y)$

Wyszy rząd (parametr  $\alpha$ ) - gęstsze przebiegi miedzi, pikselami; większe złożoności obliczeniowe, większe rozmycia.

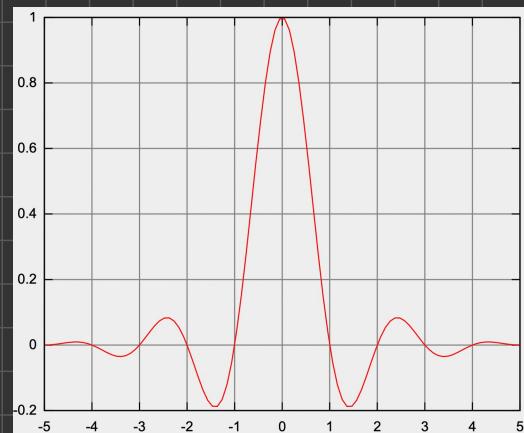
Piesel: abgeln →

2 օյնաթեր

$$L(x,y) = \frac{\sum_{i=Lx+1-a+1}^{Lx+a} \sum_{j=Ly+1-a+1}^{Ly+a} L((x,y) - (i,j)) s_{ij}}{\sum_{i=Lx+1-a+1}^{Lx+a} \sum_{j=Ly+1-a+1}^{Ly+a} L((x,y) - (i,j))}$$



Lanczos kernel for  $a=3$



Lanczos-5

Gaussian interpolation  $\rightarrow O(nm^2)$  cu2 = "gaussian", E - well-defined kernels

Używając kernela Gaussa do wyznaczania wag pikseli do zsumowania

kernel  $\rightarrow G(x,y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x-y^2}{2\sigma^2}} = \exp\left(-\frac{\|x-y\|^2}{2\sigma^2}\right)$  also  $\sigma$  - oddylene standardabweichung: Gauß

$$\text{new pixel standard deviation} \rightarrow \sqrt{\sum_{j=L+1-i+1}^{L+1+i} \sum_{k=L+1-j+1}^{L+1+k}, G(i-L+1, j-L+1)}$$

Wybór  $\sigma$  definiuje zarówno efektywną, jak i gubojącą interpolację.

## Dobre zakończenie detale

dla funkcji jednej zmiennej  $\rightarrow g(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{x^2}{2\sigma^2}}$