

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ
ÚSTAV TELEKOMUNIKÁCIÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION
DEPARTMENT OF TELECOMMUNICATIONS

INTERPOLÁCIA SIGNÁLOV POMOCOU NURBS KRIVIEK

DIPLOMOVÁ PRÁCA
MASTER'S THESIS

AUTOR PRÁCE
AUTHOR

Bc. LUBOMÍR ŠKVARENINA



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA ELEKTROTECHNIKY
A KOMUNIKAČNÍCH TECHNOLOGIÍ
ÚSTAV TELEKOMUNIKÁCIÍ

FACULTY OF ELECTRICAL ENGINEERING AND
COMMUNICATION
DEPARTMENT OF TELECOMMUNICATIONS

INTERPOLÁCIA SIGNÁLOV POMOCOU NURBS KRIVIEK SIGNAL INTERPOLATION USING NURBS CURVES

DIPLOMOVÁ PRÁCA
MASTER'S THESIS

AUTOR PRÁCE
AUTHOR

Bc. LUBOMÍR ŠKVARENINA

VEDÚCI PRÁCE
SUPERVISOR

Ing. JÁN ZÁTYIK

BRNO 2013



VYSOKÉ UČENÍ
TECHNICKÉ V BRNĚ

Fakulta elektrotechniky
a komunikačních technologií

Ústav telekomunikací

Diplomová práce

magisterský navazující studijní obor
Telekomunikační a informační technika

Student: Bc. Ľubomír Škvarenina

ID: 115286

Ročník: 2

Akademický rok: 2013/2014

NÁZEV TÉMATU:

Interpolace signálů pomocí NURBS křivek

POKYNY PRO VYPRACOVÁNÍ:

Nastudujte a popište princip známých metod interpolace obrazů. Teoreticky popište interpolační metody a nejméně čtyři z nich implementujte v programovém prostředí MATLAB.

Dále nastudujte některé z pokročilejších metod interpolace obrazu, zejména metodu interpolačních NURBS křivek. Dané metody implementujte v programové prostředí MATLAB a vytvořte jednotné grafické, nebo textové rozhraní pro všechny naprogramované funkce.

Dále popište některé metody subjektivního a objektivního ohodnocení výsledků, pomocí kterých se srovná kvalita výstupních obrazů. Získané výstupy z implementovaných interpolačních metod porovnejte pomocí vybraných metod subjektivního a objektivního ohodnocení.

DOPORUČENÁ LITERATURA:

[1] PIEGL,L., TILLER,W.: The NURBS Book, Monographs in Visual Communications. Springer, Berlin, 1997

[2] ŽÁRA,J., BENEŠ,B., FELKEL,P.: Moderní počítačová grafika, Computer Press, Praha, 2004

Termín zadání: 10.2.2014

Termín odevzdání: 30.5.2014

Vedoucí práce: Ing. Ján Zátyik

Konzultanti diplomové práce:

doc. Ing. Jiří Mišurec, CSc.

Předseda oborové rady

UPOZORNĚNÍ:

Autor diplomové práce nesmí při vytváření diplomové práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č.40/2009 Sb.

ABSTRAKT

Diplomová práca sa zaoberá problematikou interpolácie obrazu. Cieľom práce je nastudovať a následne teoreticky popísať charakter jednotlivých metód interpolácia obrazu a niektoré z nich implementovať v programovom prostredí MATLAB[®]. Úvodná časť tejto práce teoreticky priblíži dôležité pojmy, ktoré úzko súvisia s danou problematikou spracovania digitálneho obrazu postačujúce pre jej principiálne pochopenie. V nasledujúcej časti diplomovej práce budú rozobrané všetky dnes bežne využívané metódy interpolácie obrazu. Pojednávať sa bude predovšetkým o metóde interpolácie obrazu pomocou najbližšieho suseda a interpoláciách za pomocou polynómov ako je (bi)lineárna, (bi)kvadratická, (bi)kubická metóda. Následne práca teoreticky rozoberá teóriu jednotlivých druhov kriviek a splajnov. Konkrétnejšie speje do ich najpoužívanejšej varianty B-splajn kriviek a ich zovšeobecnenia nazývaných NURBS, spolu s riešením problému interpolácie týmito krivkami. Záverečnú kapitolu tvoria výsledky dosiahnuté v programovom prostredí MATLAB[®].

KLÚČOVÉ SLOVÁ

obraz, interpolácia, aproximácia, prevzorkovanie, rekonštrukcia, polynom, krivka, bézier, b-splajn, NURBS

ABSTRACT

Diploma thesis deals with image interpolation. The aim of this work is to study theoretically and then describe the nature of the various methods of image interpolation and some of them implemented in the program MATLAB[®]. The introductory part of this work theoretically closer to important terms that are closely related to this topic of digital image processing sufficient to understand the principle. In the following of the thesis will be discussed all of today's commonly used method of image interpolation. Will hear all about the method of image interpolation using nearest neighbor interpolation and image help of polynomials such as (bi)linear, (bi)quadratic and (bi)cubic method. Then work theoretically analyzes the theory of individual species curves and splines. More specifically, coming to their most frequently used variants of B-spline curves and their generalizations called NURBS, with addressing the problem of interpolating these curves. The final chapter consists of the results achieved in the program MATLAB[®].

KEYWORDS

image, interpolation, approximation, resampling, reconstruction, polynomial, curve, bezier, b-spline, NURBS

ŠKVARENINA, Lubomír *Interpolácia signálov pomocou NURBS kriviek*: diplomová práca. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav telekomunikací, 2013. 91 s. Vedúci práce bol Ing. Ján Zátyik

PREHLÁSENIE

Prehlasujem, že som svoju diplomovú prácu na tému „Interpolácia signálov pomocou NURBS kriviek“ vypracovala samostatne pod vedením vedúceho diplomovej práce, využitím odbornej literatúry a ďalších informačných zdrojov, ktoré sú všetky citované v práci a uvedené v zozname literatúry na konci práce.

Ako autor uvedenej diplomovej práce ďalej prehlasujem, že v súvislosti s vytvorením tejto diplomovej práce som neporušil autorské práva tretích osôb, najmä som nezasiahol nedovoleným spôsobom do cudzích autorských práv osobnostných alebo majetkových a som si plne vedomý následkov porušenia ustanovenia § 11 a nasledujúcich autorského zákona č. 121/2000 Sb., o právu autorskom, o právach súvisejúcich s právom autorským a o zmene niektorých zákonov (autorský zákon), v znení neskorších predpisov, vrátane možných trestnoprávných dôsledkov vyplývajúcich z ustanovenia časti druhej, hlavy VI. diel 4 Trestného zákoníka č. 40/2009 Sb.

Brno

.....

(podpis autora)

POĎAKOVANIE

Rád by som poďakoval vedúcemu diplomovej práce pánovi Ing. Jánovi Zátyikovi za odborné vedenie, konzultácie, trpezlivosť a podnetné návrhy k práci.

Brno

.....

(podpis autora)



Faculty of Electrical Engineering
and Communication
Brno University of Technology
Technická 12, CZ-61600 Brno
Czech Republic
<http://www.six.feec.vutbr.cz>

POĎAKOVANIE

Výskum popísaný v tejto diplomovej práci bol realizovaný v laboratóriách podporených z projektu SIX; registračné číslo CZ.1.05/2.1.00/03.0072, operačný program Výskum a vývoj pre inovácie.

Brno

.....

(podpis autora)



EVROPSKÁ UNIE
EVROPSKÝ FOND PRO REGIONÁLNÍ ROZVOJ
INVESTICE DO VAŠÍ BUDOUCNOSTI



OBSAH

Úvod	12
1 Úvod do počítačovej grafiky	13
1.1 Rastrová grafika	13
1.2 Vektorová grafika	13
1.3 Ľudský systém videnia (HVS)	14
1.3.1 Ľudské oko	15
1.4 Farebný model	16
1.4.1 Model RGB	17
1.4.2 Model CMY	18
1.4.3 Model HSB	19
1.4.4 Model HLS	19
1.4.5 Stupne šedi	20
2 Digitálne spracovanie obrazu	22
2.1 Digitálny obraz	22
2.2 Digitalizácia obrazových signálov	23
2.2.1 Vzorkovanie	24
2.2.2 Kvantovanie	25
2.3 Reprezentácia digitálneho obrazu	25
2.4 Fourierov obraz	27
2.4.1 Spojitá Fourierová transformácia	28
2.4.2 Diskrétna Fourierová transformácia	28
2.4.3 Shannonov vzorkovací teorém	29
2.4.4 Konvolúcia	29
2.4.5 Aliasing	29
3 Analytické krivky v rovine	31
3.1 Modelovanie kriviek polynómami	31
3.1.1 Vlastnosti kriviek	33
3.1.2 Krivky po častiach polynomicke	34
3.2 Spojitosť kriviek	34
3.3 Splajn krivky	36
4 Rekonštrukcia a interpolácia digitálneho obrazu	37
4.1 Interpoláčn� metody	38
4.2 Ideálna interpolácia	39
4.3 Sinc interpolácia	40

4.4	Interpolácia najbližším susedom (NN)	41
4.5	Lineárna a bilineárna interpolácia	42
4.6	Kvadratická a bikvadratická interpolácia	44
4.7	Kubická a bikubická interpolácia	44
4.8	Fraktálna interpolácia	45
4.9	Zhrnutie	46
4.9.1	Rozdiely v metódach	46
4.9.2	Neadaptívne a adaptívne metódy	48
5	B-splajn, NURBS	49
5.1	Predchodcovia B-splajn	49
5.1.1	Fergusonove (Hermitovské) krivky	49
5.1.2	Bézierové krivky	50
5.1.3	Coonsove krivky	52
5.2	B-splajn, NURBS krivky	53
5.2.1	B-splajn bazové funkcie	53
5.2.2	B-splajn krivky	53
5.2.3	NURBS krivky	54
5.2.4	Problém a riešenie interpolácie NURBS	55
6	Metódy hodnotenia podobnosti obrazu	58
6.1	Objektívne metódy hodnotenia	58
6.1.1	Stredná kvadratická chyba (MSE)	58
6.1.2	Odstup signálu od šumu (SNR)	58
6.1.3	Špičkový odstup signálu od šumu (PSNR)	59
6.1.4	Štruktúrna podobnosť (SSIM)	59
6.2	Subjektívne metódy hodnotenia	60
6.2.1	Dvojitá stimulovaná miera zhoršenia (DSIS)	60
6.2.2	Jednoduchá stimulácia (SS)	60
7	Porovnanie metód v programe MATLAB®	61
7.1	Programové riešenie interpolácie	61
7.1.1	Interpolácia funkciou <code>interp1</code>	61
7.1.2	Interpolácia funkciou <code>interp2</code>	62
7.1.3	Interpolácia využitím toolboxu <code>msplines1</code>	63
7.1.4	Interpolácia využitím toolboxu <code>B-spline tools 1.2</code>	63
7.2	Subjektívne porovnanie metódou DSIS	64
8	Záver	67

Literatúra	69
Knižnice programu MATLAB®	73
Zoznam symbolov, veličín a skratiek	74
A Výsledné testované obrazy MATLAB	77

ZOZNAM OBRÁZKOV

1.1	Rastrová a vektorová grafika.	13
1.2	Spektrum elektromagnetického žiarenia.	14
1.3	Prierez ľudského oka.	15
1.4	Skladanie farieb.	16
1.5	Reprezentácia farebných modelov RGB a CMY v jednotkovej kocke.	17
1.6	Farebné modely HSB a HLS.	20
2.1	Šedo-tónový obraz 512×512 a priestorové znázornenie $z = f(x, y)$	23
2.2	Spojité obraz následne po vzorkovaní a kvantovaní.	24
2.3	Reprezentácia pixelov v obraze.	26
2.4	Vznik aliasu.	30
3.1	Krivky s radiacimi bodami.	32
3.2	Krivka $Q(t)$ z dvoch segmentov $Q_1(t)$ a $Q_2(t)$	35
3.3	Parametrická a geometrická spojitosť.	36
4.1	Jednorozmerný rozklad 2D 4×4 interpolácie bodu (x, y)	39
4.2	Konvolučné jadro $^{\text{Sinc}}h_N(x)$	40
4.3	Konvolučné jadro $^{\text{Sused}}h_1(x)$	41
4.4	Konvolučné jadro $^{\text{Linear}}h_2(x)$	42
4.5	Bilineárna interpolácia jasu. Prebrané z [28]	43
4.6	Konvolučné jadro $^{\text{Kvadr}}h_3(x)$	44
4.7	Konvolučné jadro $^{\text{Kubic}}h_4(x)$	45
4.8	Bikubická interpolácia jasu a funkcie g_1, g_2, g_3, g_4 . Prebrané z [28]	46
4.9	Ilustrácia 2D interpolačných metód.	47
4.10	Porovnanie jednotlivých 1D interpolácií.	48
5.1	B-splajn funkcie stupňa $n = 0, 1, 2, 3$ centrované pre $t = 0$	54
7.1	Schéma spracovania obrazu pomocou <code>interp1</code>	62
7.2	Schéma spracovania obrazu pomocou <code>interp2</code>	62
7.3	Schéma spracovania obrazu použitím <code>msplines1</code>	63
7.4	Schéma spracovania obrazu použitím <code>B-spline tools 1.2</code>	64
7.5	Graf výsledkou metódy DSIS pre obrázok lena.png.	64
7.6	Graf výsledkou metódy DSIS pre obrázok barbara.png.	65
7.7	Graf výsledkou metódy DSIS pre obrázok flinstones.png.	65
7.8	Graf výsledkou metódy DSIS pre obrázok zoneplate.png.	66
A.1	Časť testovacích obrázkov pre demonštráciu.	77
A.2	Obrázok lena.png.	78
A.3	Obrázok barbara.png.	80
A.4	Obrázok zoneplate.png.	82
A.5	Obrázok flinstones.png.	84

A.6	Obrázok fingerprint.png.	86
A.7	Obrázok gradient.png.	88
A.8	Obrázok cameraman.png.	90

ÚVOD

Interpolácia sa v princípe zaoberá problémom veľmi úzko spojeným s modernou počítačovou grafikou. V úvodnej kapitole sú načrtnuté a rozobraté určité pojmy, ktoré v širšej miere súvisia s počítačovou grafikou. Najskôr sú priblížené pojmy vychádzajúce z vnímania a prístupu ku grafike, popísaný je princíp ľudského videnia a následne používané farebné modely.

Druhá kapitola rozoberá digitálny obraz, spôsob a prístup k jeho spracovaniu. Približuje digitalizáciu obrazových signálov pozostávajúcu zo vzorkovania a kvantovania. Ďalej definuje samotný spôsob reprezentácie digitálneho obrazu v počítačových systémoch. Následne je priblížený Fourierov obraz spolu s Fourierovou transformáciou ako spojitou, tak i diskretnou. Objasnené sú pojmy ako Shannonov vzorkovací teorém, konvolúcia či aliasing.

Tretia kapitola sa zaoberá analytickými krivkami v rovine a ich vyjadreniami. Rozoberá samotné vyjadrenia kriviek pomocou polynómov a následne definuje ich požadované vlastnosti. Neopomína ani spôsob nadväzovania kriviek definovaných daným druhom spojitosti.

Nasledujúca kapitola už rozoberá spomínanú interpoláciu a rekonštrukciu digitálneho obrazu. Prierez jednotlivých interpolačných metód tvorí sinc interpolácia, interpolácia najbližším susedom, (bi)lineárna, (bi)kvadratická a (bi)kubická interpolácia. Okrajovo je spomenutá ešte fraktálna interpolácia a na koniec kapitoly tvorí zhrnutie daných metód.

Ďalšia kapitola konečne pristupuje k definícii B-splajn a NURBS kriviek. Najskôr sa priblížia predchodcovia B-splajn kriviek ako Fergusonove, Béziérové a Coonsove krivky. Ďalej sa už definujú obecné bázové funkcie z ktorých vychádzajú B-splajn krivky a ich zovšeobecnenie nazývané NURBS. Zavŕšením je popis problému interpolácie pomocou NURBS a jeho riešenia.

V predposlednej časti sú definované ešte metódy hodnotenia kvality spracovávaného obrazu. Pojednáva sa o objektívnych metódach ako stredná kvadratická chyba (MSE), odstup signálu od šumu (SNR), špičkový odstup signálu od šumu (PSNR) a štrukturálna podobnosť (SSIM). Pre úplnosť sú ešte priblížené subjektívne metódy hodnotenia obrazu podľa doporučenia ITU-R BT.500-11.

Posledná časť uvádza riešenia súvisiace s danou problematikou naprogramované v programovom prostredí MATLAB®.

1 ÚVOD DO POČÍTAČOVEJ GRAFIKY

V počítačovej grafike existujú dva základné prístupy zobrazovania, ukladania a spracovania obrazovej informácie. Primárne rozdelenie spočíva v rastrovej a vektorovej grafike.

1.1 Rastrová grafika

Typickým charakterom rastrovej grafiky je, že vždy ju tvorí konečný súbor digitálnych hodnôt nazývaných obrazové body či pixely¹. Dvojmerný obraz je reprezentovaný pomocou tzv. rastrovej (pravouhlej) mriežky, kde každý jednotlivý pixel predstavuje hodnotu svetelnej intenzity. Keďže charakter rastrového obrazu presne určuje polohu každého pixelu, je možné vyjadriť všetky svetelné intenzity v jednotlivých bodoch obrazu ako diskretnú maticu vzoriek. Práve z toho dôvodu sa pre rastrovú grafiku zaužíval všeobecný termín „bitmapa“, ktorý je skrátčením slovného spojenia bitová mapa.



Obr. 1.1: Rastrová a vektorová grafika.

Nespornou výhodou je značne verné uchovanie snímku, akým je fotografia charakterizovaná spojitou funkciou (napr. jasom). Hlavnou nevýhodou popri pamäťovej náročnosti je viditeľný raster, a následne i obmedzenia z neho vyplývajúce, ako je strata obrazovej kvality pri zmene veľkosti. Z tejto skutočnosti úzko spätnej s rastrovým obrazom, prichádza čiastočné riešenie použitím interpolácie obrazu opísanej v kap. 4.

1.2 Vektorová grafika

Vektorová grafika má svoj základ v matematických výpočtoch a zobrazeniach použitím vektorov. Obraz je reprezentovaný základnými geometrickými útvarmi so

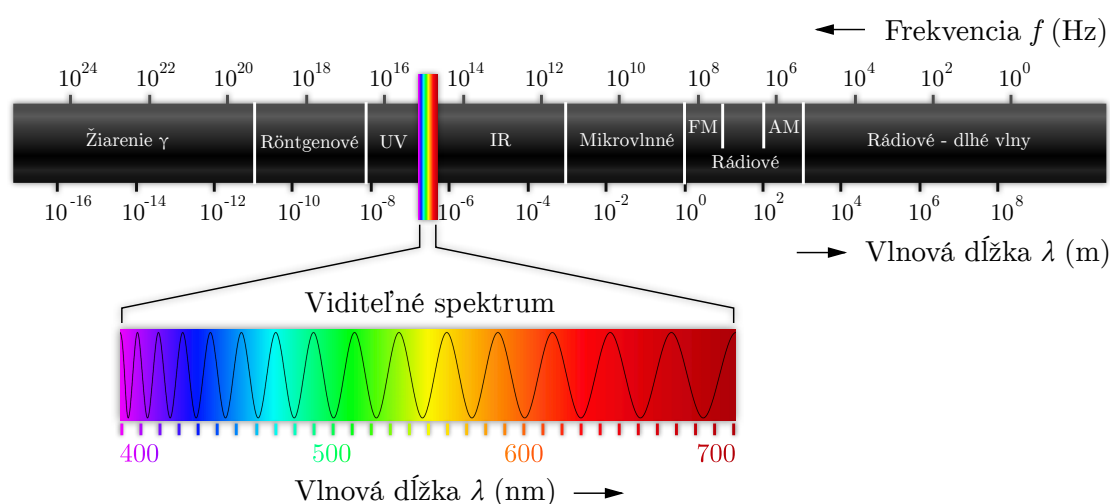
¹*Pixel* (z angl. slov „picture element“) je termín užívaný pre základnú stavebnú jednotku počítačovej rastrovej grafiky.

špecifikovanou priestorovou pozíciou ako sú body, priamky, krivky či mnohoúhelníky. Nie je zložený z jednotlivých bodov usporiadaných do pravouhlej mriežky ako pri rastrovej grafike. Už v samotnom princípe sa práca s vektorovou grafikou značne odlišuje od práce s grafikou rastrovou.

Značnou výhodou je nielen ľubovoľné zväčšovanie či zmenšovanie obrazu bez straty kvality, ale i možnosť pracovať s každým objektom oddelene. Taktiež je výhodou menšia pamäťová náročnosť umožňujúca vyšší stupeň kompresie. Oproti rastrovej grafike je nevýhodou vektorovej grafiky nemožnosť spracovávať spojitý snímok, akým je fotografia.

1.3 Ľudský systém videnia (HVS)

Ľudský systém videnia (z ang. *Human Visual System*) vychádza z prirodzených vlastností vnímania farieb a svetla pomocou ľudského oka. Svetlo, ktoré je človek schopný vnímať zrakom, predstavuje iba viditeľnú časť z celého elektromagnetického (EM) spektra² zobrazeného na obr. 1.2.



Obr. 1.2: Spektrum elektromagnetického žiarenia.

Elektromagnetické žiarenie vzniká osciláciou elektricky nabitých materiálov, čím nadobúda charakter vlnenia šíriacim sa po priamych dráhach s rýchlosťou približne 3×10^8 km/s. Viditeľnú časť spektra pre ľudské oko tvorí kontinuálna oblasť vlnových dĺžok v rozmedzí 380 až 720 nm, v ktorej vnímame žiarenia s určitou vlnovou dĺžkou ako farbu[34].

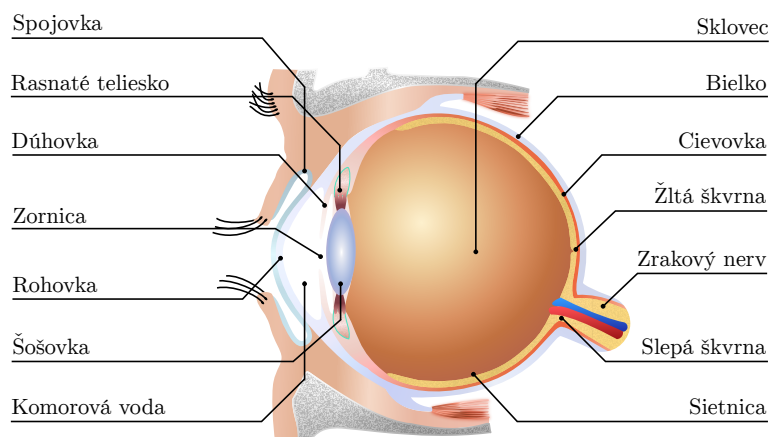
²Pojmom *spektrum* sa označuje celý rozsah energetických hladín (vlnových dĺžok), ktoré nadobúdajú fotóny pri svojom šírení priestorom a časom [12].

Svetelný zdroj (slnko, žiarovka) vysiela lúče všetkých frekvencií v danom pásme, ktoré sa skladajú vo výsledné biele svetlo označované ako „achromatické“. Pri dopade bieleho svetla na objekt sú niektoré frekvencie pohltené a niektoré zase odrazené. To, čo vníma človek ako farebný tón, je v skutočnosti kombinácia frekvencií prítomných v odrazenom svetle s dominantnou vlnovou dĺžkou [34]. Avšak nie každej farbe prislúchala iná vlnová dĺžka. Je to z dôvodu, že ľudské oko podľa intenzity dopadajúceho svetla rozlišuje nie len farbu, ale i ďalšie podnety, akými sú:

- *Jas* (I, intensity), prípadne *svietivosť* (luminosity), odpovedá intenzite svetla. Vyššia intenzita znamená jasnejší zdroj.
- *Sýtosť* (S, saturation) udáva čistotu farby svetla. Užšie frekvenčné spektrum značí vyššiu čistotu.
- *Svetlosť* (B, brightness) určuje veľkosť achromatickej zložky vo svetle s určitou dominantnou frekvenciou.

1.3.1 Ľudské oko

Ľudské oko je podobné guli o priemere približne 20 mm. Následný rez oka je možné vidieť na obr. 1.3 spolu s jednotlivými popiskami jeho hlavných častí³. Cez šošovku sa zaostrený obraz vonkajšieho svetla premieta na sietnicu, ktorú tvorí tenká fotocitlivá vrstva, pokrývajúca približne dve tretiny vnútorného povrchu oka. Sietnica obsahuje dva druhy fotocitlivých receptorov → *tyčinky* (rods) a *čapíky* (cones). V ľudskom oku



Obr. 1.3: Prierez ľudského oka.

je približne 120×10^6 tyčiniek a 8×10^6 čapíkov. Pri nízkych hladinách osvetlenia

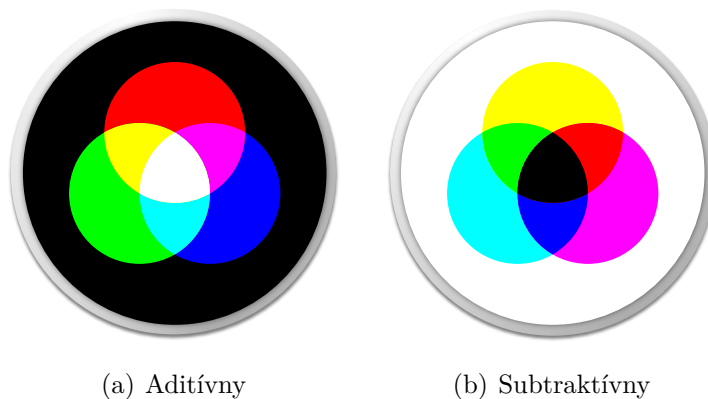
³Človek má prirodzene trojrozmerné binokulárne videnie, čo je videnie oboma očami zároveň. Znamená to, že obrazy vnímané simultánne oboma očami sa spoja v jeden čo umožňuje vnímať hĺbku priestoru [3].

slúžia tyčinky, ktoré sú približne $10\times$ citlivejšie než čapíky. Čapíky tvoria základ farebného videnia a reagujú na vyššie hodnoty intenzity osvetlenia. Každý čapík obsahuje jeden z troch druhov fotopigmentov, ktoré sa líšia podľa relatívnej citlivosti na vlnové dĺžky svetla. Farebné pigmenty čapíkov sú zastúpené nerovnomerne a to 64 % červené, 32 % zelené a 2 % modré pigmenty [34].

1.4 Farebný model

Farebný model je abstraktná matematická štruktúra popisujúca reprezentáciu farieb ako n -tícu čísiel v digitálnom obraze. Takto je možné použitím troch či štyroch hodnôt farebných zložiek definovať rozsah farieb zobrazovaných v danom režime. Od tejto množiny farieb sa odvíja mutácia nazývaná farebný priestor, kedy je farebný model doplnený o presný popis, ako majú byť jednotlivé zložky interpretované (napr. pozorovacie podmienky)[4]. Používaným termínom, ktorý štandardizuje v rámci farebného priestoru základný rozsah farieb býva tzv. gamut⁴.

Ako už bolo vysvetlené v kap. 1.3, farba je daná kombináciou frekvencií svetla rôznych vlnových dĺžok. Rôzne farebné modely sa snažia túto vlastnosť napodobiť čo najvernejšie. Základný spôsob miešania farieb je *aditívny* (súčtový) na obr. 1.4(a) a *subtraktívny* (rozdielový) na obr. 1.4(b). Ako je zrejmé z obr. 1.4, pri aditívnom



Obr. 1.4: Skladanie farieb.

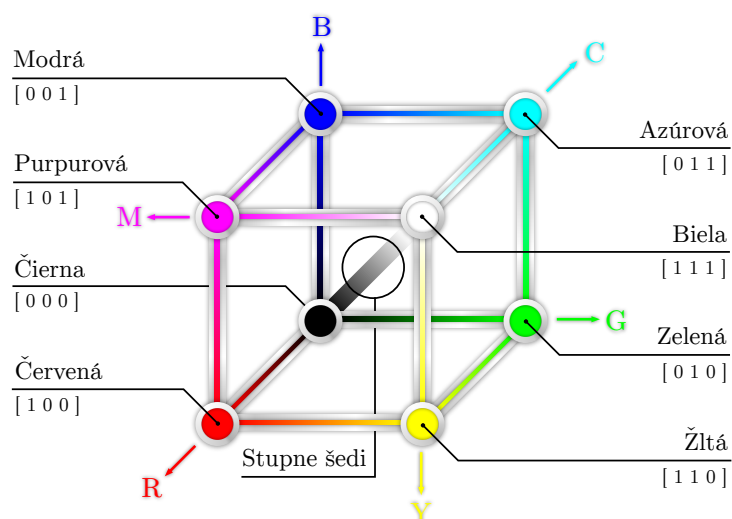
skladaní farieb pridaním jednotlivých farebných zložiek vytvoríme svetlo väčšej intenzity. Naproti tomu, pri subtraktívnom skladaní farieb pridaním jednotlivých farebných zložiek uberáme časť pôvodnej intenzity svetla.

⁴ *Gamut* určuje reálne dosiahnuteľnú oblasť farieb v určitom farebnom priestore. Farby mimo túto oblasť je možné v danom farebnom priestore zobrazovať len približne.

1.4.1 Model RGB

Model RGB je tvorený kombináciou troch základných farieb z farebného spektra určujúce i jeho názov. Využíva práve aditívny spôsob skladania farieb. Farebný model RGB bol štandardizovaný už v roku 1931 komisiou CIE⁵. Na typickej farebnej obrazovke vidíme farebný obraz ako výsledok troch kanálov, ktorými sú *červená* (R, red), *zelená* (G, green) a *modrá* (B, blue) farba. Týmto jednotlivým zložkám boli priradené nasledujúce vlnové dĺžky → červená = 700 nm, zelená = 546,1 nm, modrá = 435,8 nm [8].

Takto je možné farbu vyjadriť trojicou (vektorom určujúcim farbu), ktorej zložky nadobúdajú hodnoty z intervalu $\langle 0, 1 \rangle$, uvádzané v celočíselnom rozsahu 0 – 255. Toto odpovedá kódovaniu každej zo zložiek RGB v jednom bajte⁶. Najbežnejším spôsobom je vyjadrenie farebných zložiek 3 bajtami, kedy každý jednotlivý kanál je možné zakódovať 8 bitmi s 256 úrovňami sýtosti. Počet farebných odtieňov je teda $256^3 = 16777216$.



Obr. 1.5: Reprezentácia farebných modelov RGB a CMY v jednotkovej kocke.

Farebný model RGB predstavuje jednotková kocka, ako je vidieť na obr. 1.5. V jej počiatku $[0, 0, 0]$ sa nachádza farba *čierna* (K, black), zatiaľ čo vrchol $[1, 1, 1]$ tvorí farba *biela* (W, white). Základné farby ležia na vrcholoch v osiach a zostávajúce vrcholy určujú farby doplnkové, z ktorých vychádza model CMY kap. 1.4.2. V prípade, že všetky tri farebné zložky zobrazíme v plnej intenzite, tak na výstupnom zariadení bude výslednou farbou biela. Podobným spôsobom však môžeme získať

⁵ CIE (z angl. slov „International Commission on Illumination“) je medzinárodnou organizáciou venujúcou sa svetlu, osvetľovaniu, farbe a kolorimetrickým sústavám.

⁶ Bajt (z angl. slova „byte“) je jednotkou digitálnej informácie v počítačových systémoch. Používaným synonymom je „oktet“ z dôvodu, že 1 bajt je rovný 8 bitom.

stupne šedi (hlavná diagonála medzi čiernou a bielou farbou) kap. 1.4.5, skladaním všetkých troch farieb so zhodnou postupne klesajúcou intenzitou. Ak hodnoty základných farebných zložiek sú normalizované v intervale $\langle 0, 1 \rangle$, tak pre doplnkové farby platí

$$B + G = C, \quad R + B = M, \quad R + G = Y, \quad R + G + B = W. \quad (1.1)$$

Model RGBA

Farebný priestor RGBA, presnejšie $RGB\alpha$, vychádza z modelu RGB a je rozšírený o tzv. α -kanál obsahujúci informáciu o priehľadnosti. Tým nemení pôvodný farebný priestor modelu RGB, ale pridáva k nemu iba ďalšiu informáciu. Ak je pre každý farebný kanál RGB vyhradený 1 bajt, tak v tom prípade je α -kanál zakódovaný taktiež v 8 bitoch a dosahuje rovnako 256 úrovní (priehľadnosti, nie sýtosti).

1.4.2 Model CMY

Tento model vychádza z ľudskej skúsenosti miešania farieb v reálnom prirodzenom prostredí, kedy tmavšia farba vzniká práve pridávaním pigmentu. V prípade CMY modelu sa teda jedná o subtraktívny (analógia prekryvania farieb u maliarskych či tlačiarenských techník) spôsob skladania farieb, kedy pridávaním farby uberáme na pôvodnej intenzite. Taktiež vyplývajú z názvu tohoto modelu, je charakterizovaný farbami označovanými ako *azúrová* (C, cyan), *purpurová* (M, magenta) a *žltá* (Y, yellow). Rovnako ako RGB vychádza z jednotkovej kocky, ktorá je znázornená na obr. 1.5. Jedná sa ale o presný opak, kedy vrchol $[0, 0, 0]$ odpovedá farbe bielej a vrchol $[1, 1, 1]$ farbe čiernej. Rovnako pre výpočet doplnkových farieb v modeli CMY platia v prípade hodnôt normalizovaných v intervale $\langle 0, 1 \rangle$ nasledujúce vzťahy

$$Y + M = R, \quad Y + C = G, \quad C + M = B, \quad C + M + Y = K. \quad (1.2)$$

Prevod medzi modelmi RGB a CMY je v princípe jednoduchou záležitosťou, keďže sú vzájomné voči sebe inverzné. Postačuje ak si farebný priestor RGB vyjadríme trojprvkovou maticou $[r, g, b]$ a vektor $[c, m, y]$ v priestore CMY získame odčítaním matíc:

$$\begin{bmatrix} c & m & y \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 \end{bmatrix} - \begin{bmatrix} r & g & b \end{bmatrix}. \quad (1.3)$$

Tento spôsob skladania farieb je využívaný predovšetkým pri tlačiarňach, keďže pridávaním pigmentov dostávame tmavšiu farbu, ktorej podklad tvorí biely papier. Naproti tomu RGB bol výhodný pre monitory, ktorých tienidlo je čiernej farby a rozsvietením každého farebného kanálu pixelu sa dosiahlo väčšej intenzity.

Model CMYK

Nedokonalé krytie jednotlivých farebných zložiek modelu CMY v reálnych podmienkach, kedy pri ich zmiešaní nikdy nevznikne dokonalo čierna farba, ale len špinavá šedivo-hnedá farba, dalo vznik rozšírenému modelu CMYK⁷. Pre tento hlavný dôvod je žiadúce používať čierny pigment, vrátane využitia k stmaveniu jednotlivých odtieňov farieb. I z ekonomického hľadiska je výroba čierneho pigmentu ďaleko menej nákladná, ako ju dosahovať zmesou pigmentov troch farieb.

1.4.3 Model HSB

Model HSB pozostáva z troch základných parametrov (zložiek), ktorými sú *tón/odtieň* (H, hue), *saturácia/sýtosť* (S, saturation) a *jas* (B, brightness)⁸. Model odpovedá kuželu, prípadne i šesťbokému ihlanu na obr. 1.6(a). Prednosťou je predovšetkým intuitívnejšie miešanie farieb oproti predchádzajúcim modelom z pohľadu ľudského vnímania.

Pri manipulácii s farbou sa najskôr v prvej fáze volí príslušný farebný odtieň H, v druhej fáze nasýtenie S spolu so svetlosťou B. Výhodou modelu je, že farebný odtieň sa pri manipulácii s nasýtením a svetlosťou nemení, čo sa dá pri miešaní farieb v RGB či CMYK dosiahnuť obtiažne. Analógiu v praxi tvorí práca s vodovými farbami. Kde najskôr volíme farebný pigment a následne ho riedime vodou, čiže uberáme nasýtenie a nakoniec ztmavujeme či zosvetľujeme pridávaním čiernej alebo belosti, čiže upravujeme svetlosť [2].

Farebný tón H určujúci hlavnú spektrálnu zložku, predstavuje farba odrazená alebo prechádzajúca objektom. Meria sa ako poloha na klasickom farebnom kruhu (0° až 360°). Obecne sa odtieň označuje názvom farby. Sýtosť S (niekedy označovaná ako *chroma*) určujúca silu alebo čistotu farby, predstavuje množstvo šedi v pomere k odtieňu H na achromatickej ose. Meria sa v percentách od 0% (šedá) do 100% (plne sýta farba). Na farebnom kruhu vzrastá sýtosť od stredu k okrajom. Hodnota jas B alebo množstvo bieleho svetla určuje relatívnu svetlosť, alebo tmavosť farby. Jas teda vyjadruje, koľko svetla farba odráža.

1.4.4 Model HLS

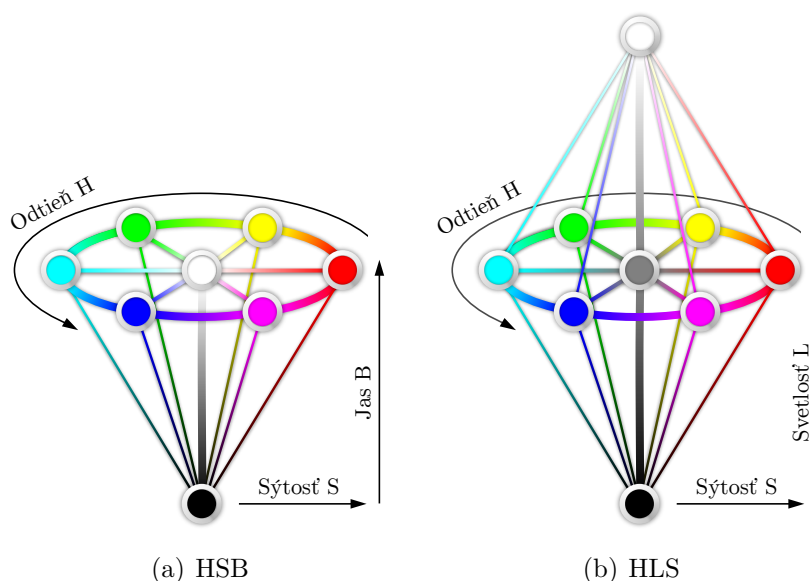
Model HLS pozostáva taktiež z troch parametrov (zložiek), ktorými sú ako v predchádzajúcom modeli *tón/odtieň* (H, hue), *saturácia/sýtosť* (S, saturation), avšak tretiou zložkou je tzv. *svetlosť* (L, lightness). Tento model tvorí dvojité rotačný

⁷Doplnkové písmeno *K* pochádza správne z angl. slova *key*, čo značí „kľúčovú farba“ a nie čiernu zo slova *black*.

⁸Ekvivalentným označením býva i HSV (Hue, Saturation, Value).

kužel znázornený na obr. 1.6(b). HLS je značne podobný modelu HSB a podarilo sa mu odstrániť jeden z nedostatkov tohoto modelu, ktorým je absencia symetrie z hľadiska jasú. Sýtosť S leží na vodorovnej ose, svetlosť L (poloha medzi čiernou a bielou farbou) na zvislej ose a farebný tón H predstavuje uhlová hodnota.

Vzhľadom k tomu, že schopnosť rozlišovania farebných odtieňov skutočne klesá so stmavením a zosvetlením základnej čistej farby, zvyšovanie a znižovanie svetlosti farby skutočne spočíva v pridávaní svetlého alebo tmavého pigmentu, tak tento model najviac odpovedá skutočnosti. Problém oboch uvedených modelov tvorí posun po kružnicu pre zmenu parametru H, kde vykazujú nespojitosť na vlnovej dĺžke farby pri prechode modrou a fialovou farbou.



Obr. 1.6: Farebné modely HSB a HLS.

1.4.5 Stupne šedi

Stupne šedi je možné označiť ako jednorozmerný model obsahujúci výhradne *desaturované* farby tvoriace diagonálu na jednotkovej kocke obr. 1.5. Desaturáciou sa myslí určenie jasú z priemeru zložiek r, g, b v modeli RGB. V prípade, že je potrebné rôznofarebný obraz previesť na obraz v stupňoch šedi, nie je možné výslednú intenzitu získať prostým súčtom intenzít jednotlivých kanálov. Ľudské oko je totiž rôzne citlivé na každú farebnú zložku (najcitlivejšie je na zeleno-žltú [34]). Z toho dôvodu je potrebné priradiť jednotlivým kanálom rôznu váhu. Do stupňov šedi sa preto obraz prevádza podľa nasledujúceho empirického vzťahu

$$I(r, g, b) = 0,299r + 0,587g + 0,114b, \quad (1.4)$$

kde pro každý pixel sa spočíta výsledná intenzita jasů.

2 DIGITÁLNE SPRACOVANIE OBRAZU

Digitálne spracovanie obrazu, ako vedeckotechnická disciplína spadajúca do spracovania signálu, sa zaoberá spracovávaním digitálnych 2D signálov v počítačových systémoch. Pri digitálnom spracovávaní obrazu je najčastejším cieľom spracovávať predovšetkým obrazy, ktoré vznikajú ako obrazy spojité.

2.1 Digitálny obraz

Obraz môže byť definovaný ako dvojrozmerná funkcia $f(x, y)$, kde x a y sú rovinné súradnice, a f v každom páre súradníc (x, y) predstavuje svetelnú intenzitu. Pri formálnom vymedzení použijeme matematický model, ktorým je spojitá funkcia dvoch premenných - *obrazová funkcia* [13].

Z matematického hľadiska je *analógový obraz* zobrazením $f : \mathbb{R}^2 \rightarrow \mathbb{R}^n$. Ak obraz uvažujeme v modeli stupňov šedi $n = 1$, uvedené je možné chápať ako reálnu funkciu

$$z = f(x, y), \quad z \in \mathbb{R}, \quad (2.1)$$

demonštrovanú na obr. 2.1. V modeloch s viac zložkami bude dimenzia $n > 1$, pre RGB model konkrétne $n = 3$, čo je možné zapísať

$$\mathbf{z} = f(x, y), \quad \mathbf{z} \in \mathbb{R}^3. \quad (2.2)$$

Jedná sa o zobrazenie so spojitým časom $x, y \in \mathbb{R}$ [26].

Keďže spracovávaný obraz ma obmedzené rozmery, je možné definičný obor obrazovej funkcie zapísať ako karteziánsky súčin dvoch spojitých intervalov z oboru reálnych čísiel

$$f : (\langle x_{min}, x_{max} \rangle \times \langle y_{min}, y_{max} \rangle) \rightarrow H. \quad (2.3)$$

Reálne čísla x, y , kde $x : x_{min} \leq x \leq x_{max}$ a $y : y_{min} \leq y \leq y_{max}$ sú súradnice bodu v dvoch rozmeroch, v ktorých funkcia nadobúda nejakej hodnoty z z oboru hodnôt, tj. $z \in H$ [34].

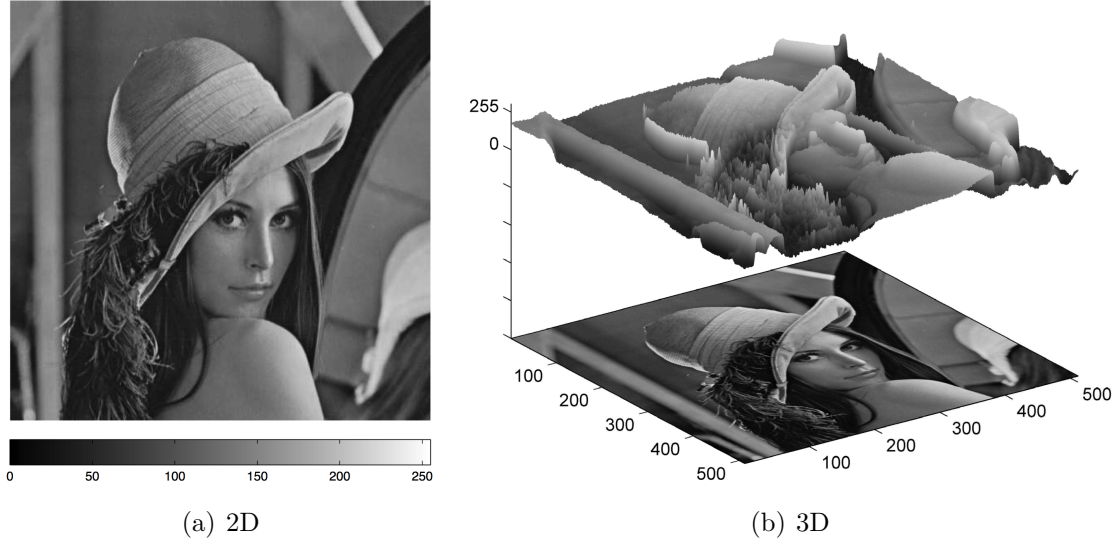
Pri práci s obrazom v počítačovej grafike je zriedka k dispozícii spojitý definičný obor obrazovej funkcie. Preto v prípade *diskrétneho obrazu* bude obraz formálne chápaný ako zobrazenie

$$f : \{1, 2, \dots, N_x\} \times \{1, 2, \dots, N_y\} \rightarrow H_1 \times H_2 \times \dots \times H_m, \quad (2.4)$$

kde H_m predstavuje množinu hodnôt, ktoré môže nadobúdať m -tá zložka modelu (*kanál*), $m = 1, \dots, n$. Obraz sa chápe ako n matic, každá rozmeru $N_x \times N_y$ [26].

Digitálny obraz o rozmeroch 512×512 zobrazený na obr. 2.1(a) v stupňoch šedi pri 8 bitoch bude mať tvar

$$f : \{1, 2, \dots, 512\} \times \{1, 2, \dots, 512\} \rightarrow \{0, 1, \dots, 255\}. \quad (2.5)$$



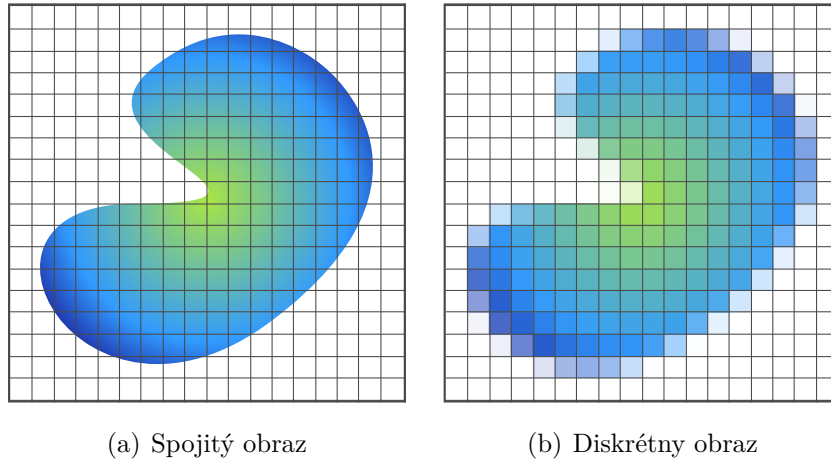
Obr. 2.1: Šedo-tónový obraz 512×512 a priestorové znázornenie $z = f(x, y)$.

2.2 Digitalizácia obrazových signálov

Nato aby bolo vôbec možné nejaký obraz v počítačových systémoch spracovávať, je potrebné ho najskôr nasnímať. Tento obraz, ktorý je potrebné nasnímať má však teoreticky nekonečný rozsah obrazových hodnôt či už je to vo veľkosti alebo vo farbe. Avšak v počítači bude zobrazený v konečnom množstve pixelov a v konečnom množstve farieb. To nastoľuje hlavný problém, ktorý spočíva práve v reálnej predlohe obrazu, ktorá je spojitá, naproti čomu obraz reprezentovaný v počítačových systémoch je vždy diskretný. Spojitá funkcia jednej premennej bude označovaná ako $f(x)$ a jej diskretný obraz ako I_i , ktorú je možno zobecniť na spojitú funkciu dvoch premenných $f(x, y)$ a jej odpovedajúca diskretná funkcia $I_{i,j}$.

Nato, aby sme prešli od spojitaj 2D funkcie $f(x, y)$ k diskretnaj 2D funkcií $I_{i,j}$, je potrebné urobiť proces prechodu nazývaný *digitalizácia*. Tento proces v sebe zahŕňa dva podprocesy, ktorými sú *vzorkovanie* a *kvantovanie*. Taktiež je možné povedať, že digitalizácia hodnôt súradníc je vzorkovaním a digitalizácia hodnôt amplitúdy je kvantovaním.

Proces digitalizácie je možné vidieť na obr. 2.2. Spojitý obraz, premietnutý na rovinu sensorového poľa na obr. 2.2(a), je následne vzorkovaný a kvantovaný vo výsledný diskretný obraz na obr. 2.2(b).



Obr. 2.2: Spojitý obraz následne po vzorkovaní a kvantovaní.

Pre úplnosť je potrebné uviesť, že po spracovaní do digitalizovanej podoby môže vzniknúť taktiež požiadavka, aby bol obraz rekonštruovaný do pôvodnej spojitaj podoby. Tomuto sa následne v tejto práci venuje celá kap. 4.

2.2.1 Vzorkovanie

Vzorkovaním (z ang. „Sampling“) spojitaj 1D funkcie $f(x)$ rozumieme zaznamenávanie hodnôt (vzoriek) v dopredu daných intervaloch. Diskretnu 1D funkciu získanú pravidelným vzorkovaním budeme označovať ako I_i a vzdialenosť dvoch vzoriek označíme ako Δx . Pôvodná spojitá funkcia môže byť definovaná na ľubovoľnom intervale $x \in \langle x_0, x_1 \rangle$ a vzorky budeme indexovať nasledujúcim spôsobom:

$$I_i = f(x_0 + i\Delta x); i = 0, 1, \dots \quad (2.6)$$

Pri vzorkovaní logicky musí dochádzať k strate informácie. Pri predpoklade, že funkcia je konštantná a došlo k jej prudkej zmene v okamžiku $f(x_0 + \frac{\Delta x}{2})$ pri vzorkovanom kroku Δx , hodnote $f(x_0)$, odpovedá vzorka I_0 a $f(x_0 + \Delta x)$ vzorka I_1 . Presnejšie $f(x_0) = I_0$ a $f(x_0 + \Delta x) = I_1$ [34]. Hodnota v polovici intervalu s týmto predpokladom jednoducho v rade výstupných vzoriek zaznamenaná nie je. Takto o udalosti medzi výstupnými vzorkami chýba informácia. Tento spôsob vzorkovania, kedy je hodnota odobraná v jednom bode, sa nazýva bodové vzorkovanie (*point sampling*). Technicky je najbežnejšie vzorkovanie, kedy je zaznamenaná reprezentácia hodnôt celého vzorkovaného intervalu Δx . Tejto metóde sa hovorí plošné vzorkovanie (*area sampling*), kedy hodnota vzorku je určená napríklad priemerom zo

všetkých hodnôt

$$I_i = \frac{1}{\Delta x} \int_{x_0+i\Delta x}^{x_0+(i+1)\Delta x} f(t) dt. \quad (2.7)$$

Plošné vzorkovanie vyžaduje vyššiu výpočtovú náročnosť než vzorkovanie bodové. Pri definovaní vzdialenosti dvoch vzoriek ako Δx rozumieme *vzorkovaciu frekvenciu* f_s ako hodnotu

$$f_s = \frac{1}{\Delta x} \quad (2.8)$$

a jednotkou je počet vzoriek za jednotku času (Hz) či na jednotku vzdialenosti (dpi) [34].

Zovšeobecnenie jednorozmerného vzorkovania 2.7 do dvoch rozmerov možno zapísať ako

$$I_{i,j} = \frac{1}{\Delta x \Delta y} \int_{x_0+i\Delta x}^{x_0+(i+1)\Delta x} \int_{y_0+j\Delta y}^{y_0+(j+1)\Delta y} f(p, q) dp dq. \quad (2.9)$$

2.2.2 Kvantovanie

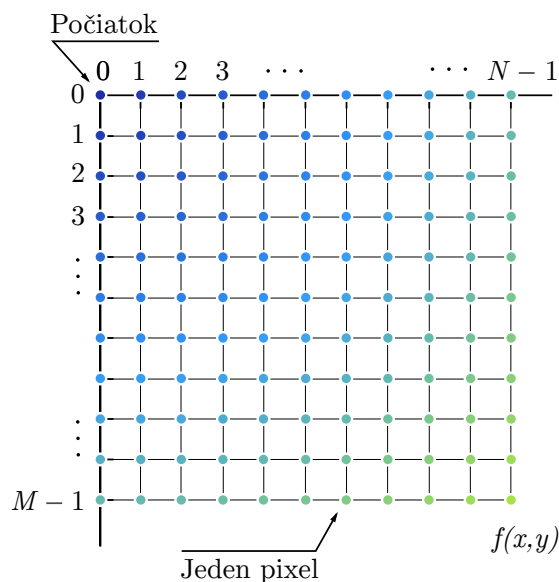
Pri praktickej reprezentácii signálu v počítačových systémoch, je často potrebné uchovávať jednotlivé vzorky signálov v premenných s obmedzeným rozsahom (bajt, slovo atd.). V takýchto prípadoch realizujeme prevod skutočných úrovní signálu na hodnoty, ktoré je možné v premennej zvoleného typu uchovávať, čo nazývame *kvantovaním* (z ang. „Quantization“). Na konci reťazca digitálneho spracovania je naopak často potrebné obraz na základe jeho kvantovanej reprezentácie opäť zrekonštruovať tak, aby čo najlepšie odpovedal obrazu teoretickému [28].

Kvantovanie prebieha v oboru hodnôt obrazovej funkcie, ktorá sa rozdelí na intervaly s jedinou pridelenou zástupnou hodnotou. Podľa spôsobu rozdelenia kvantovanej veličiny hovoríme o kvantovaní *uniformnom* alebo *neuniformnom*. Uniformné kvantovanie používa konštantnú dĺžku intervalu, zatiaľ čo neuniformné kvantovanie využíva premenlivú dĺžku intervalu. Neuniformné kvantovanie sa využíva zriedkavo i cez svoju možnosť zohľadniť nerovnomerné rozloženie hodnôt meranej veličiny. Pri kvantovaní samozrejme rovnako dochádza ku strate informácie, keďže množina hodnôt je nahradená hodnotou jedinou. Túto stratu označuje pojem *kvantizačná chyba* a v počítačovej grafike sa prejavuje napríklad na plochách s malou zmenou gradientu ako náhly skok farieb [34].

2.3 Reprezentácia digitálneho obrazu

Výsledkom vzorkovania a kvantovania je matica reálnych čísiel. Predpokladajme, že obraz $f(x, y)$ je vzorkovaný tak, aby výsledný digitálny obraz mal M riadkov a N

stĺpcov. Hodnoty súradníc (x, y) sa teraz stali diskretnými kvantitami. Pre prehľadnosť a pohodlie budeme používať celočíselné hodnoty týchto diskretných súradníc, čo napríklad značí, že počiatok obrazu je $(x, y) = (0, 0)$. Ďalšie hodnoty súradníc



Obr. 2.3: Reprézntácia pixelov v obraze.

pozdĺž prvého riadku obrazu sú reprezentované ako $(x, y) = (0, 1)$, pričom je dobré mať na pamäti, že sa jedná o druhú vzorku pozdĺž prvej rady. Samozrejme to neznamená, že sa jedná o skutočne hodnoty fyzikálnych súradníc, kedy bola odobratá vzorka obrazu. Pre ilustráciu postačí obr. 2.3.

Podľa predchádzajúceho zobrazenia môžeme zapísať kompletný $M \times N$ digitálny obraz v nasledujúcom kompaktnom maticovom tvare

$$f(x, y) = \begin{bmatrix} f(0, 0) & f(0, 1) & \cdots & f(0, N - 1) \\ f(1, 0) & f(1, 1) & \cdots & f(1, N - 1) \\ \vdots & \vdots & \ddots & \vdots \\ f(M - 1, 0) & f(M - 1, 1) & \cdots & f(M - 1, N - 1) \end{bmatrix}. \quad (2.10)$$

Pravá strana rovnice je samozrejme definíciou digitálneho obrazu, v ktorom každý element tohoto maticového pola je nazývaný pixel. Niekedy býva vhodnejšie použiť tradičnejší zápis matice digitálneho obrazu s jeho prvkami

$$\mathbf{A} = \begin{bmatrix} a_{0,0} & a_{0,1} & \cdots & a_{0,N-1} \\ a_{1,0} & a_{1,1} & \cdots & a_{1,N-1} \\ \vdots & \vdots & \ddots & \vdots \\ a_{M-1,0} & a_{M-1,1} & \cdots & a_{M-1,N-1} \end{bmatrix}. \quad (2.11)$$

Z čoho je jasne definované, že $a_{i,j} = f(x = i, y = j) = f(i, j)$. Tento proces digitalizácie vyžaduje určenie hodnôt pre L , označujúci diskkrétne hodnoty stupňov šedi pre každý pixel. Pre hodnoty M a N nie sú kladené žiadne iné nároky okrem toho, aby sa jednalo o kladné celé čísla. Hodnotu L pre počet stupňov šedi je možné vyjadriť ako

$$L = 2^k. \quad (2.12)$$

Predpokladáme, že jednotlivé úrovne sú rovnomerne rozložené, a taktiež sú celými číslami v intervale $\langle 0, L-1 \rangle$. Niekedy rozsah hodnôt rozloženia šedo-tónovej stupnice sa nazýva ako dynamický rozsah určujúci, akú významnú časť stupnice jednotlivé hodnoty pokrývajú. Máme tým na mysli rozloženie medzi najsvetlejším a najtmavším bodom stupnice. Snímok majúci veľký dynamický rozsah bude mať vysoký kontrast, a naopak snímok s nízkym dynamickým rozsahom bude vykazovať matný až sivý vzhľad. Veľkosť v bitoch b potrebných na uloženie digitalizovaného obrazu, môžeme vypočítat ako

$$b = M \times N \times k. \quad (2.13)$$

V prípade rovnostranného štvorcového obrazu kedy $M = N$ môžeme zjednodušiť na

$$b = N^2 k. \quad (2.14)$$

Ak obraz môže mať 2^k úrovni šedi, je bežnou praxou odkazovať sa na obraz ako „ k -bitový“. Napríklad obraz s 256 možnými úrovňami šedi sa nazýva ako 8 bitový obraz [13].

2.4 Fourierov obraz

Fourierova transformácia slúži k prevodu obrazu do duálneho priestoru, ktorý zjednodušuje niektoré operácie s obrazom a je vhodný pre pochopenie niektorých javov a vlastností obrazu. V tomto prípade sa používajú dve reprezentácie obrazu. Prvou je obraz reprezentovaný diskkrétou maticou pixelov, kde sa táto reprezentácia označuje ako *priestorová oblasť*. V teórii signálov pod pojmom *oblasť časová*. Fourierov obraz je reprezentáciou obrazu vo *frekvenčnej oblasti*, kde je zložený z nekonečne mnoho sínusových signálov, ktoré majú rôznu amplitúdu a sú rôzne fázovo posunuté. Analógiou k tomu je reprezentácia zvuku ako signálu, ktorý sa skladá z hĺbok (signálov nízkej frekvencie) a výšok (signálov vysokej frekvencie). Čím silnejšie sú hĺbky, tým väčšia je amplitúda nízkych frekvencií, a naopak, čím silnejšie sú výšky, tým väčšia je amplitúda vysokých frekvencií. Šum, ktorý je obyčajne nežiadúci sa prejavuje ako vysoké frekvencie. A nežiadúci nízkofrekvenčný signál je nazývaný *alias*, ktorého prejav v obraze je nazvaný *aliasing* [34], rozobraný v kap. 2.4.5.

2.4.1 Spojitá Fourierová transformácia

Výpočet, pri ktorom získame Fourierov obraz funkcie reprezentovanej v časovej oblasti, sa nazýva *dopredná Fourierová transformácia*, a pôvodný obraz získame z Fourierovho obrazu *spätnou Fourierovou transformáciou*. Fourierovým obrazom 1D funkcie $f(x)$ definovanej na spojitom definičnom obore budeme rozumieť komplexnú funkciu $F(u)$. Získame ju integráciou doprednej Fourierovej transformácie

$$F(u) = \int_{-\infty}^{+\infty} f(x)e^{-i2\pi ux} dx, \quad (2.15)$$

kde $i = \sqrt{-1}$ je komplexná jednotka. Spätná Fourierová transformácia komplexnej integrovateľnej funkcie $F(u)$, ktorá realizuje prechod od frekvenčnej oblasti k priestorovej je definovaná vzťahom

$$f(x) = \int_{-\infty}^{+\infty} F(u)e^{+i2\pi ux} du. \quad (2.16)$$

Fakt, že $f(x)$ a $F(u)$ sú dvojakým vyjadrením rovnakej funkcie zviazanej reláciou Fourierovej transformácie, býva niekedy zvykom ich označovať ako

$$f(x) \iff F(u). \quad (2.17)$$

Pre zovšeobecnenie Fourierovej transformácie pre 2D obraz spojitej funkcie $f(x, y)$, označíme Fourierov obraz ako $F(u, v)$, z čoho analogicky dopredná a spätná transformácia má tvar

$$F(u, v) = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} f(x, y)e^{-i2\pi(ux+vy)} dx dy, \quad (2.18)$$

$$f(x, y) = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} F(u, v)e^{+i2\pi(ux+vy)} du dv. \quad (2.19)$$

2.4.2 Diskrétna Fourierová transformácia

Signál s obmedzenou postupnosťou vzorkou, ktorá sa periodicky opakuje používa diskrétnu Fourierovu transformáciu. Priama a spätná *diskrétna Fourierova transformácia* (DFT) funkcie I_k je definovaná ako

$$F_n = \frac{1}{N} \sum_{k=0}^{N-1} I_k e^{-i2\pi \frac{nk}{N}}, \quad (2.20)$$

$$I_k = \sum_{n=0}^{N-1} F_n e^{+i2\pi \frac{nk}{N}}.$$

Dualitu medzi priestorovou I_k a frekvenčnou reprezentáciou F_n obrazu vyjadríme pomocou

$$I_k \iff F_n. \quad (2.21)$$

2.4.3 Shannonov vzorkovací teorém

Frekvenčne obmedzená funkcia (tzv. *bandlimited function*) má konečné amplitúdové spektrum. V konečnom amplitúdovom spektre existuje najvyššia frekvencia f_{\max} s vlastnosťou pre všetky frekvencie u , pre ktoré platí, že $u > f_{\max}$ je $|F(u)| = 0$. Amplitúda týchto frekvencií je nulová a nenesú žiadnu energiu. Najjednoduchším prípadom frekvenčne obmedzenej funkcie je samozrejme funkcia $\sin(x)$, ktorej vo Fourierovom spektre odpovedá jediný bod. Najvyššia nenulová frekvencia f_{\max} sa nazýva *Nyquistovo kritérium*. Vzorkovacia frekvencia f_s viz 2.8 a najvyššia frekvencia f_{\max} uvádza do vzťahu Shannonova vzorkovacia veta (taktiež Shannonov vzorkovací teorém), ktorý znie:

Signál spojité v čase je plne určený postupnosťou vzoriek odoberaných v rovnakých intervaloch Δx , ak je ich frekvencia $f_s = \frac{1}{\Delta x}$ väčšia než dvojnásobok najvyššej frekvencie v signále f_{\max} , tj. ak je

$$f_s > 2f_{\max}. \quad (2.22)$$

Každá frekvenčne obmedzená funkcia je reprezentovaná do určitej veľkosti rastru presne. Ďalšie zvyšovanie vzorkovacej frekvencie nevedie k pridaniu detailov a je teda zbytočné. Väčšina scén v počítačovej grafike nie je frekvenčne obmedzená [34].

2.4.4 Konvolúcia

Konvolúcia je matematická operácia dvoch funkcií (spracováanej funkcie s konvolučným jadrom), ktorej výsledkom je upravená pôvodná funkcia. Patrí medzi najzákladnejšie spôsoby spracovávania diskrétného 2D obrazu v počítačových systémoch. Označovaná býva symbolom $*$. Pri práci s digitálnym signálom je využívaná diskrétna konvolúcia, ktorá pre výstupný obraz ma tvar

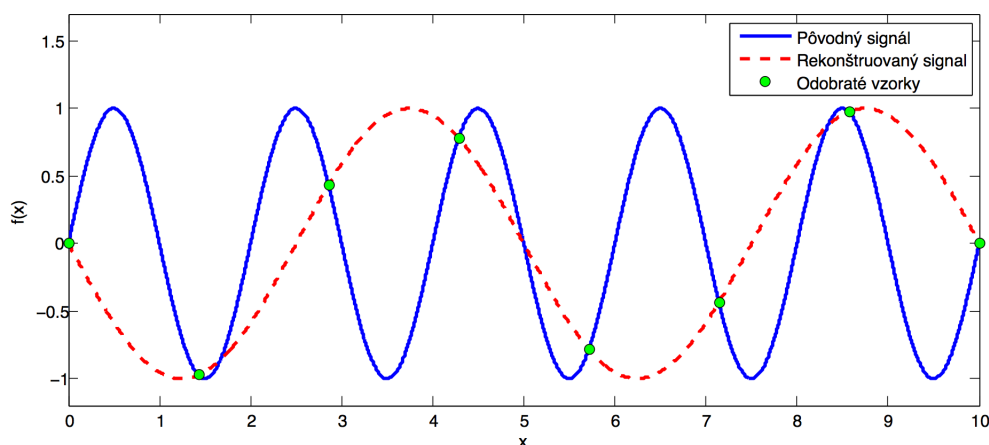
$$I'_{i,j} = I_{i,j} * h_{i,j} = \sum_{x=-k}^k \sum_{y=-k}^k I_{i-x,j-y} h_{i,j}. \quad (2.23)$$

Funkcia $h_{i,j}$ je konvolučným 2D jadrom, ktoré môžeme chápať ako okno (*konvolučnú masku*) o rozmeroch $\langle -k, k \rangle \times \langle -k, k \rangle$, posúvané na príslušné miesta v obraze [34]. Každý pixel prekrytý oknom vynásobíme koeficientom v príslušnej bunke a urobíme súčet všetkých týchto hodnôt, čím dostaneme jeden nový pixel.

2.4.5 Aliasing

Počítačový obraz s diskrétnou povahou je najčastejšie získavaný vzorkovaním spojitaj scény, a preto sa stretávame s nevyhnutným problémom nazývaným aliasing. Alias vzniká pri rekonštrukcii signálu vzorkovaného pod Nyquistovým limitom. Prejavuje sa ako nová nízkofrekvenčná informácia, ktorá nebola v pôvodnom signále

prítomná. Prakticky sa jedná o prípad, kedy pôvodná funkcia obsahuje details, ktoré nie je možné v raste zobrazit'. Tieto štruktúry s vysokou frekvenciou nie je možné vzorkovaním postihnúť. Vzniká v dvoch prípadoch. Prvý prípad je, keď pôvodná funkcia je frekvenčne obmedzená. Čo znamená, že neexistuje žiadna maximálna frekvencia a funkciu nie je možné v diskkrétnej mriežke rastru reprezentovať presne. Druhý prípad je, keď pôvodná funkcia je frekvenčne obmedzená. Čo znamená, že v jej Fourierovom spektre existuje určitá maximálna frekvencia f_{\max} , ale funkciu vzorkujeme s frekvenciou pod Nyquistovým limitom ($< 2f_{\max}$). Je možné mu predísť odfiltrovaním vysokých frekvencií, či rozmazaním obrazu pred zmenšením. Antialiasing je proces, ktorý sa snaží minimalizovať alias, či zubaté diagonálne hrany nazývané *jaggies*. Ďalším nepríjemným efektom vychádzajúcim z aliasingu pri zmenšovaní obrazu je vznik efektu *moaré*¹.



Obr. 2.4: Vznik aliasu.

Prvou a najdôležitejšou skutočnosťou je potrebné si uvedomiť, že k zachyteniu drobnej kresby a malých detailov je potrebné mať dostatočný počet vzoriek z originálnej predlohy. Nepomôžu ani vzorky získané interpoláciou. Interpolácia chýbajúce vzorky odhaduje na základe akéhosi obecného zjednodušeného modelu. Drobné individuálne details, ktoré na pôvodnom vzorku nie sú zachytené, nemôže v žiadnom prípade doplniť. Zmenšenie obrázku je preto obecné nevratná zmena spôsobujúca, že prípadne spätné zväčšenie na pôvodnú veľkosť už nezíska drobné details.

¹Moaré (z franc. slova *moiré*) je rušivý efekt vznikajúci prekryvaním alebo interferenciou dvoch pravidelných a veľmi málo odlišných rastrov (lísiacich sa fázou, orientáciou prípadne kmitočtom).

3 ANALYTICKÉ KRIVKY V ROVINE

Krivky sú obyčajne v počítači reprezentované ako sústava parametrov nejakej rovnice, ktorá je následne generatívne zobrazovaná. Využitie kriviek má širokú škálu od modelovania v 2D rovine či 3D priestore, pre definíciu dráhy pohybujúcich sa objektov, až po modelovanie tvaru znakov vo fontoch. Existujú tri základné analytické vyjadrenia kriviek:

- *explicitné* $\rightarrow y = f(x)$,
- *implicitné* $\rightarrow F(x, y) = 0$,
- *parametrické* \rightarrow bodová rovnica $Q(t) = [x(t), y(t)]$,
 \rightarrow vektorová rovnica $\mathbf{q}(t) = (x(t), y(t))$.

V explicitnom tvare je možné vyjadriť iba krivky, ktoré sú zároveň funkciami, tzn. že v hodnote x z definičného oboru odpovedá jediná funkčná hodnota y . Implicitné vyjadrenie je pomerne obtiažne zobraziteľné pre nemožnosť postupného výpočtu krivky v obecných prípadoch. Najčastejším vyjadrením krivky v počítačovej grafike býva práve parametrický tvar. Súradnice sú funkciami parametra času t , z intervalu $t \in \langle t_{\min}, t_{\max} \rangle$ a najčastejšie je volený z $t \in \langle 0, 1 \rangle$. Jeho výhodou je predovšetkým závislosť súradníc krivky výhradne na parametre t (času). Vďaka tomu je možné vyjadriť krivky prechádzajúce viackrát (v rôznych časových okamžikoch) rovnakými bodmi na ploche.

Vektor $\mathbf{q} = Q(t) - [0, 0]$ sa nazýva *polohový vektor* a jeho veľkosť je rovná vzdialenosti bodu $Q(t)$ od počiatku.

Dotyčnicový vektor v bode $Q(t_0)$ je určený deriváciami parametricky vyjadrenej krivky po zložkách v tvare

$$\mathbf{q}'(t_0) = (x'(t_0), y'(t_0)) = \left(\frac{dx(t_0)}{dt}, \frac{dy(t_0)}{dt} \right). \quad (3.1)$$

Rovnica dotyčnice sa vypočíta z dotyčnicového vektoru a bodu na krivke ako

$$P(u) = Q(t_0) + u\mathbf{q}'(t_0). \quad (3.2)$$

Vektor \mathbf{q}' sa taktiež nazýva *smerový vektor priamky* a využíva sa pri nadväzovaní kriviek a skladaní zložitejších tvarov [34].

3.1 Modelovanie kriviek polynómami

Polynóm (mnohočlen) tvorí súčet alebo rozdiel jednočlenov. V počítačovej grafike ako základný druh parametrických kriviek tvoria krivky polynomické

$$Q_n(t) = a_n t^n + a_{n-1} t^{n-1} + \dots + a_1 t + a_0, \quad (3.3)$$

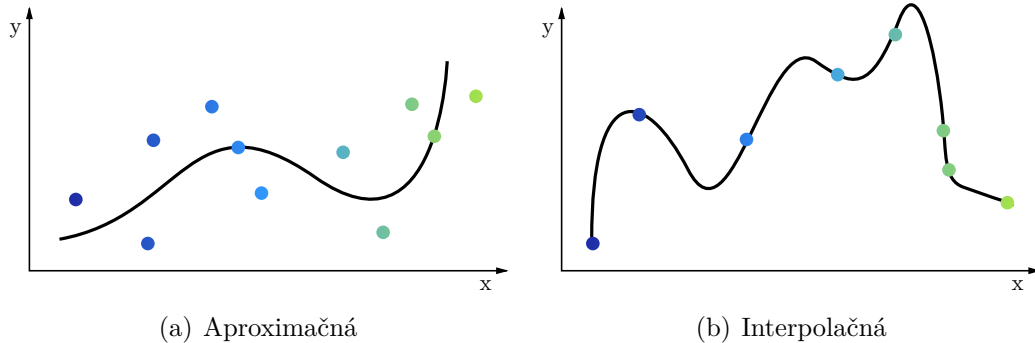
ktoré je možno maticovo zapísať ako

$$Q_n(t) = \begin{bmatrix} t^n & t^{n-1} & \cdots & t & 1 \end{bmatrix} \cdot \begin{bmatrix} a_n & a_{n-1} & \cdots & a_1 & a_0 \end{bmatrix}^\top, \quad (3.4)$$

kde reálne čísla $a_n, a_{n-1}, \dots, a_1, a_0 \in \mathbb{R}$; $a \neq 0$ sú koeficienty polynómu a celé nezáporné číslo n stupeň polynómu. Rovnicu 3.4 je možné chápať ako lineárnu kombináciu jednotlivých polynómov t^n, t^{n-1} až t^0 , ktoré sú lineárne nezávislé. To znamená, že musia tvoriť bázu vektorového priestoru všetkých polynómov, ktoré sú rádu n alebo nižšieho. Takýto ľubovoľný polynóm je možné získať pomocou tejto báze a súradníc polynómu a_n, a_{n-1}, \dots, a_1 v tejto báze [26]. Polynomické krivky sú ľahko vyčísliteľné a jednoducho diferencovateľné.

Pre modelovanie krivky sa definuje niekoľko *riadiacich bodov* (*riadiaci polygón*), ktorých matematický aparát z ich polohy určí priebeh krivky. Zadávaním kriviek pomocou dotyčnicových vektorov je možné zaručiť spojitosť a hladkosť nadväzovania. Tvar krivky $Q(t)$ jednoznačne definujú riadiace body značené \mathbf{P}_i , prípadný zápis ako $Q(t) = Q(\mathbf{P}_0, \mathbf{P}_1, \dots, \mathbf{P}_n, t)$. Dotyčnicový vektor ku krivke je obecné označený ako $\mathbf{q}'(t)$ a jeho druhá derivácia $\mathbf{q}''(t)$. Dotyčnicové vektory v riadiacich bodoch \mathbf{p}'_i alebo uzloch \mathbf{q}'_i , ktorými krivka prechádza [34].

Základné delenie kriviek je na *aproximačné* a *interpolačné*. Krivka generovaná pri interpolácii musí prebiehať danými bodmi, pri aproximácii nemusí.



Obr. 3.1: Krivky s riadiacimi bodmi.

Pre 2D priestor je parametrický tvar krivky, konkrétnejšie kubiky, vyjadrený dvoma súradnicami

$$\begin{aligned} x(t) &= a_x t^3 + b_x t^2 + c_x t + d_x \\ y(t) &= a_y t^3 + b_y t^2 + c_y t + d_y, \end{aligned} \quad (3.5)$$

čo je možné zkrátene prepísať do maticového tvaru ako

$$Q(t) = \mathbf{TC} = \begin{bmatrix} t^3 & t^2 & t & 1 \end{bmatrix} \begin{bmatrix} a_x & a_y \\ b_x & b_y \\ c_x & c_y \\ d_x & d_y \end{bmatrix}. \quad (3.6)$$

Dosadením zvoleného časového okamžiku t do rovnice (3.6) získame polohu bodu $Q(t)$ na krivke a výsledný tvar kubiky určuje osem parametrov matice \mathbf{C} . Pre vyhodnotenie polohy vo viacerých bodoch je potrebné dosadiť miesto jedného t všetky požadovane, čím v matici \mathbf{T} narastie počet riadkov.

Deriváciou dotyčnicového vektoru \mathbf{q}' polynómu $Q(t)$ v bode t získame deriváciu vektoru \mathbf{T} , čo v prípade kubiky bude vďaka vzťahu 3.6 dvojjložkový vektor

$$\mathbf{q}'(t) = \frac{d}{dt}Q(t) = \frac{d}{dt}\mathbf{TC} = \begin{bmatrix} 3t^2 & 2t & 1 & 0 \end{bmatrix} \mathbf{C}. \quad (3.7)$$

Ako bolo uvedené, kubika v 2D je určená ôsmimi parametrami tvoriace prvky matice \mathbf{C} , ktorých zmena však neposkytuje intuitívne ovládanie tvaru krivky. V prípade kubik sa matica \mathbf{C} rozkladá do súčinu $\mathbf{C} = \mathbf{MG}$, kde matica konštánt \mathbf{M} (nemenná pre zvolený typ krivky) je nazývaná ako *bázová matica* a štvorprvkový vektor \mathbf{G} sa nazýva *vektor geometrických podmienok* (riadiace body, dotyčnicové vektory apod.). Súčin \mathbf{TM} definuje *polynomickú bázu* (skupinu polynómov), ktorá je spoločná pre všetky krivky určitého typu. Kubika je definovaná vzťahom

$$Q(t) = \mathbf{TMG} = \begin{bmatrix} t^3 & t^2 & t & 1 \end{bmatrix} \begin{bmatrix} m_{11} & m_{12} & m_{13} & m_{14} \\ m_{21} & m_{22} & m_{23} & m_{24} \\ m_{31} & m_{32} & m_{33} & m_{34} \\ m_{41} & m_{42} & m_{43} & m_{44} \end{bmatrix} \begin{bmatrix} G_1 \\ G_2 \\ G_3 \\ G_4 \end{bmatrix}, \quad (3.8)$$

čo je rozpísať ako súčet polynómov násobených geometrickými podmienkami

$$\begin{aligned} Q(t) = & (m_{11}t^3 + m_{12}t^2 + m_{13}t + m_{14}) \cdot G_1 + \\ & (m_{21}t^3 + m_{22}t^2 + m_{23}t + m_{24}) \cdot G_2 + \\ & (m_{31}t^3 + m_{32}t^2 + m_{33}t + m_{34}) \cdot G_3 + \\ & (m_{41}t^3 + m_{42}t^2 + m_{43}t + m_{44}) \cdot G_4. \end{aligned} \quad (3.9)$$

Polynómy sa uplatňujú ako premenlivé váhy riadené parametrom t , ktorými sú geometrické podmienky násobené. Hodnota parametru t určuje, ktorá z podmienok sa viac uplatní na časť krivky v začiatku, vnútri či na jej konci. Ako príklad posluží nasledujúca krivka, zostavená z dvoch geometrických podmienok násobených bázovými polynómami prvého stupňa, ktorá je vlastne úsečkou

$$Q(t) = (1 - t) \cdot P_1 + t \cdot P_2. \quad (3.10)$$

Geometria je teda plne určená riadiacimi bodmi P_1 a P_2 a bázové polynómy určujú vplyv týchto bodov na priebeh úsečky [34].

3.1.1 Vlastnosti kriviek

Medzi často požadované vlastnosti kriviek patrí [34]:

- Invariancia k lineárnym transformáciám a projekciám, ktorá zaručuje, že otočenie či posunutie riadiaceho polygónu a následné generovanie krivky podá rovnaký výsledok ako otočenie každého bodu z vygenerovanej krivky. Prvý prípad odpovedá práci počítača a druhý predstave človeka. Invariancia znamená, že oba prípady majú rovnaké vyústenie. Matematicky zapísané do následnej rovnosti, kde symbol T zastupuje transformáciu bodu v 2D

$$\underbrace{Q(T(\mathbf{P}_0), T(\mathbf{P}_1), \dots, T(\mathbf{P}_n), t)}_{\text{počítač}} = \underbrace{T(Q(\mathbf{P}_0, \mathbf{P}_1, \dots, \mathbf{P}_n, t))}_{\text{človek}} \quad (3.11)$$

pre všetky $t \in \langle 0, 1 \rangle$ [26].

- Krivka leží v konvexnej obálke svojich riadiacich bodov.
- Zmena polohy a/alebo váhy riadiaceho bodu zmení tvar krivky iba v okolí tohoto bodu a nie celú krivku.
- Krivka má prechádzať krajnými bodmi svojho riadiaceho polygónu.

3.1.2 Krivky po častiach polynomické

Z polynomických kriviek je možné skladať *krivky po častiach polynomické* (segmenty tvoria polynomické krivky) viac v kap. 3.3. Tento prístup vychádza z toho, že nemá zmysel modelovať zložitý tvar jednej krivky pomocou polynómu n -tého rádu. Omnoho vhodnejšie je tieto tvary modelovať ako na seba nadväzujúce krivky malej zložitosti. Najčastejšie využívané krivky bývajú krivky tretieho stupňa ($n = 3$) tzv. kubiky. Výhodami je, že poskytujú dostatočne širokú škálu tvarov, nenáročný výpočet, ľahkú manipulovateľnosť a je možné zaručiť spojitost C^2 .

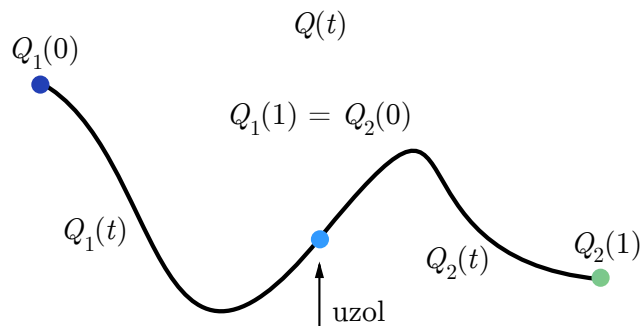
3.2 Spojitosť kriviek

Krivky môžu byť skladané z viacerých segmentov a to nielen s polynomickou reprezentáciou. Miesto, v ktorom sa krivky stýkajú sa nazýva *uzol* (*knot*). Spôsob napojenia v týchto uzloch sa označuje ako *spojitosť* (*continuity*).

Na nasledujúcom obr. 3.2 je zobrazená krivka $Q(t)$ zložená z dvoch častí $Q_1(t)$ a $Q_2(t)$, nazývaných segmenty spojených v bode $Q_1(1) = Q_2(0)$.

Parametrická spojitost

Hovoríme, že krivka $Q(t)$ patrí do triedy spojitosti C^n ak má vo všetkých bodoch spojitú deriváciu podľa parametru t do rádu n . Označenie C^n sa nazýva *parametrická spojitost stupňa n* . Dva segmenty sú spojitou nadviazané triedy C^0 , pokiaľ koncový bod prvého segmentu je počiatočným bodom segmentu druhého. Každá spojitá



Obr. 3.2: Krivka $Q(t)$ z dvoch segmentov $Q_1(t)$ a $Q_2(t)$.

krivka patrí do triedy spojitosti C^0 . Následne dva segmenty majú spojenie triedy C^1 pokiaľ dotyčnicový vektor je v koncovom bode prvého segmentu Q_1 rovný dotyčnicovému vektoru segmentu Q_2 v jeho počiatocnom bode. Pre spojitost triedy C^2 je analogicky žiadaná rovnosť prvej a druhej derivácie. Možno zapísať ako

$$\begin{aligned}
 \mathbf{q}_1(1) &= \mathbf{q}_2(0) \\
 \mathbf{q}'_1(1) &= \mathbf{q}'_2(0) \\
 \mathbf{q}''_1(1) &= \mathbf{q}''_2(0) \\
 &\vdots \\
 \mathbf{q}_1^{(n)}(1) &= \mathbf{q}_2^{(n)}(0),
 \end{aligned} \tag{3.12}$$

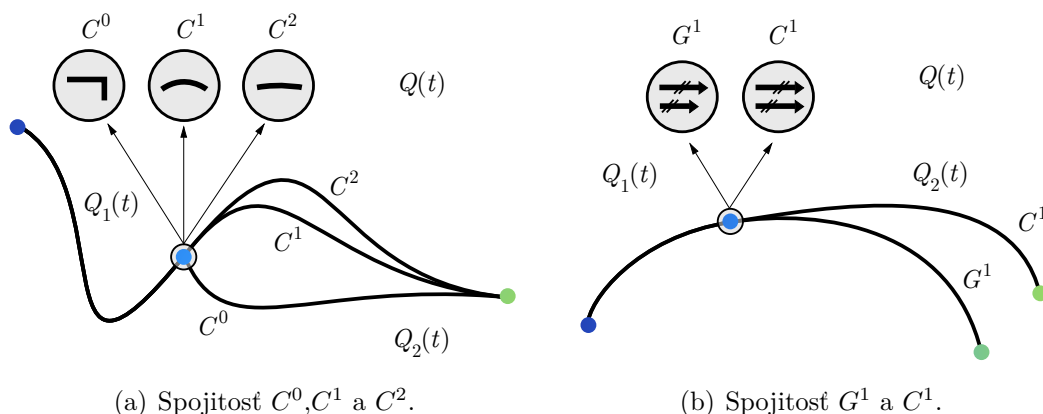
v skrátrenom tvare zapisujeme

$$\mathbf{q}_1^{(i)}(1) = \mathbf{q}_2^{(i)}(0); \quad \forall i = 0, 1, \dots, n. \tag{3.13}$$

Pre $i = 0$ sa krivky dotýkajú v koncových bodoch a pre $i = 1$ nastáva rovnosť dotyčnicových vektorov [26]. Pri spojitosti C^0 sa bod pohybuje po spojitnej dráhe, avšak v uzle môže meniť skokom smer pohybu, rýchlosť i zrýchlenie. Pri spojitosti C^1 sa skokovo nemôže meniť veľkosť rýchlosti a smer pohyb a pri spojitosti C^2 zostáva nezmenené i zrýchlenie.

Geometrická spojitost

Podobne hladkosť naväzovania kriviek môže byť podľa *geometrickej spojitosti* stupňa G^n , ak v spoločnom uzle sú vektory všetkých derivácií až do rádu n rovnobežné a súčasne majú súhlasnú orientáciu (kolineárne). V tom prípade dva segmenty krivky $Q(t)$ nadobúdajú spojitost G^0 , ak koncový bod $Q_1(t)$ je totožný s počiatocným bodom $Q_2(t)$. Pre spojitost G^1 je potrebné aby boli segmenty spojitosti G^0 a súčasne



Obr. 3.3: Parametrická a geometrická spojitosť.

dotyčnicové vektory $\mathbf{q}_1(1)$ segmentu $Q_1(t)$ a $\mathbf{q}_2(0)$ segmentu $Q_2(t)$ súhlasne kolineárne

$$\mathbf{q}_1(1) = k \cdot \mathbf{q}_2(0), \quad k > 0. \quad (3.14)$$

Jedná sa o spojitosť zaručujúcu totožnosť dotyčníc a nie dotyčnicových vektorov. Pohybujúci sa bod v uzle môže meniť skokom rýchlosť, avšak nemôže zmeniť skokom smer. Subjektívne zaručuje spojitosť G^1 skoro rovnako hladký prechod ako C^1 a býva jednoduchšie zaručiť spojitosť G^1 oproti C^1 [34].

3.3 Splajn krivky

Interpoláčny polynóm pri viacerých bodoch nadobúda istú nepriaznivú vlastnosť, že i keď prechádza zadanými bodmi mimo ne sa príliš rozkmitá. Preto sa interpoláčny polynóm nehodí k aproximácii funkcie zadanej väčším počtom bodov a je potrebné hľadať iné prostriedky. Najviac využívanou metódou, ktorá vykazuje lepšie vlastnosti pri väčšom počte bodov ako interpoláčny polynóm, je konštrukcia tzv. *splajnu*¹.

Splajn krivka stupňa n je po častiach polynomickeá krivka, ktorá je triedy C^n . Nevýhodou kriviek je, že zmenou polohy jedného definujúceho bodu sa zmení tvar celej krivky. Lineárny splajn tvoriaci polynómami prvého stupňa v hraničných bodoch na seba naväzuje spojite lomenou čiarou. Kvadratický splajn tvorený sústavou kvadratických funkcií naväzuje v zadaných bodoch, ako funkčnou hodnotou, tak i prvou deriváciou. Naproti tomu kubický splajn, tvoriaci sústavou kubických funkcií naväzuje v zadaných bodoch, ako funkčnou hodnotou, tak i prvou a druhou deriváciou. Keďže lineárny splajn vykazuje ostré hroty v zadaných bodoch a kubický splajn trpí zotrvačnosťou, v praxi sa najviac využíva splajn kubický.

¹Termín *splíne* pochádza od pružného pravítka (krivítka), ktorým sa tieto krivky modelujú.

4 REKONŠTRUKCIA A INTERPOLÁCIA DIGITÁLNEHO OBRAZU

Keď hovoríme o interpolácii obrazu je potrebné uviesť, že veľkosť digitálneho obrázku je jeho veľkosť v pixeloch, a zmeniť veľkosť znamená meniť počet pixlov. Táto zmena veľkosti digitálneho obrazu v pixeloch je zásadným zásahom do digitálneho obrazu. Vo svojej podstate sa jedná o nevratný proces, kedy dochádza k strate pôvodnej informácie. Pri zmene počtu pixlov hovoríme o odobratí vzoriek v iných bodoch nazývaným prevzorkovanie, vychádzajúc z princípov vzorkovania rozobranom v kap. 2.2. Prevzorkovanie však naráža na zásadný problém vzhľadom nato, že nie je možné získať nové hodnoty z pôvodnej scény. Spojitá predloha už totiž nie je k dispozícii. Digitalizácia obrazu sama o sebe je nevratným procesom straty informácií pôvodnej spojitej scény. Preto pre odhadnutie nových hodnôt je potrebné použiť hodnoty z už dostupných diskretných vzoriek. Tento problém dáva práve priestor klasickej matematickej úlohe nazývanej *interpolácia*.

Interpolácia je opozičným procesom k vzorkovaniu, keďže spojitý signál rekonštruje zo vzoriek na zistenie jasových hodnôt medzi známymi vzorkami. Interpolovať digitálny obraz je možné za predpokladu, že sa funkcia nebude chovať náhodne. Potrebné je funkciu obecné popísať v metóde, na základe ktorej bude možné chýbajúce vzorky odhadnúť. Najbežnejšie používané metódy tvoria funkcie zložené z úsekov polynómov nízkeho stupňa.

Hlavnou úlohou rekonštrukcie digitálneho obrazu je získanie hodnôt jasu v bodoch ležiacich mimo raster. Vzhľadom k tomu, že hodnoty medzi vzorkami nie sú definované, je potrebné ich nejakým spôsobom interpolovať. A práve rôzne spôsoby interpolácie určujú rôzne metódy, ktoré budú rozobrané v rámci nasledujúceho textu líšiac sa svojou rýchlosťou a kvalitou. Interpoláciu je možné pre jednoduchosť pojať ako vkladanie obrazových bodov do digitálneho zobrazenia založených na údajoch, ktoré sú k dispozícii.

Rekonštrukciou rozumieme prechod od diskretnej funkcie I_i , definované pre $i = 1, 2, \dots$, ku spojitej funkcií $f(x)$, $x \in \mathbb{R}$. Obor hodnôt oboch funkcií je rovnaký s predpokladom, že sa jedná o reálne čísla. Rekonštrukcia digitálneho obrazu vychádza z konvolúcie¹ signálu pomocou funkcie $h(x)$, nazývanej konvolučným jadrom, prípadne označovanou ako rekonštrukčný filter. Základnou vlastnosťou funkcie je, že vymedzená plocha musí byť jednotková

$$\int_{-\infty}^{+\infty} h(t) dt = 1. \quad (4.1)$$

¹Konvolúcia je obecná interpolácia.

Ak by hodnota integrálu presahovala veľkosť 1, nová funkcia by bola konvolúciou zosilnená. V prípade hodnôt nižšej ako 1 by bola zoslabená. Pri rekonštruovaní spojitú funkcie $f(x)$ násobíme hodnotu vzoriek v bodoch I_i konvolučným jadrom, ktoré presunieme do bodu $i\Delta x$. Výsledkom sú vo smeru osi y zmeritkované konvolučné jadrá posunuté v smere osi x . Takto získame na konci spojitú funkciu $f(x)$ súčtom funkcií jednotlivých konvolučných jadier. Práve podľa typu jednotlivých konvolučných jadier popisovaných nižšie v kap. 4.1, získavame rôzne interpolovanú pôvodnú funkciu, kde jej kvalita rastie s výpočtovou náročnosťou.

Ako už bolo spomenuté, pôvodnú funkciu zo vzoriek nejde obecné získať, a ani interpolácia nenavýši množstvo primárne získanej informácie. Jedinou, zrovna nie častou výnimkou, sú frekvenčne obmedzené funkcie vzorkované s frekvenciou rovnou či vyššou danou Shannonovým teorémom. Pokiaľ nie je táto podmienka splnená, je každá rekonštrukcia viacmenej len podarenou interpoláciou funkcie pôvodnej. Keďže pôvodnú funkciu nie je možné obecné získať, je potrebné klásť určité požiadavky na interpolačnú funkciu vedúce k vizuálne prijateľným výsledkom. V prvom rade by rekonštrukčný filter nemal byť nespojitý, pretože zrekonštruovaný obraz bude obsahovať nespojitosti prejavujúce sa ako nežiadúce efekty hrán. Rekonštrukčná funkcia by ďalej mala zachovávať spojitost prvých derivácií a rekonštrukčný filter by nemal mať príliš veľkú zmenu gradientu v okolí vzorku [34].

4.1 Interpoláčn  metody

Prevzorkovanie zahrnuje v sebe dva podprocesy, ktorými s  rekonštrukcia a vzorkovanie. Cieľom je vytvoriť spojit  funkciu z diskretnych vzoriek a n sledne z nových vzoriek vytvoriť nov  rekonštruovan  obraz. Rekonštrukcia po  astiach spojit  funkcie $f(x)$ je ziskana z diskretnych hodn t ako line rna kombin cia vstupn ch d t a rekonštruk n ho jadra. Pre interpolovanie vzoriek v 1D to m žeme definovať ako

$$f(x) = \sum_i I_i \cdot h(x - i), \quad (4.2)$$

kde I_i s  diskretn  vzorkovan  hodnoty a $h(x)$ je rekonštruk n  jadro [10].

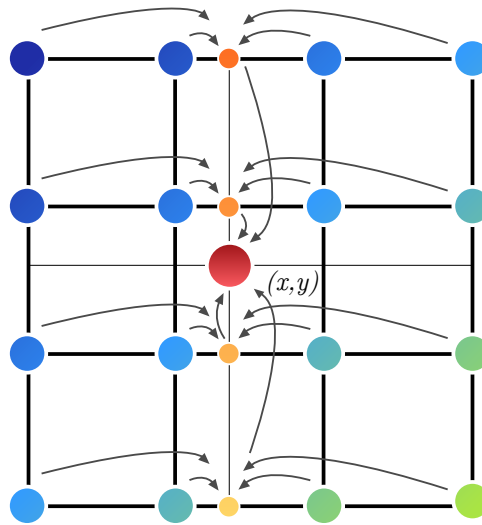
Pri prevzorkovan  digit ln ho obrazu je v sak potrebn  interpol ciou rekonštruovať dvojrozmern  spojit  sign l $f(x, y)$ z jeho diskretnych vzoriek $I_{i,j}$, kde $f, x, y \in \mathbb{R}$ a $i, j \in \mathbb{N}^0$. Tak e amplituda pre ka d  polohu (x, y) mus  byť odhadnut  z ostatn ch susedov. Form lne to m žeme op sať ako konvol ciu diskretnych vzoriek obrazu so spojitou 2D impulznou odozvou ${}_{2D}h(x, y)$ 2D rekonštruk n ho filtru

$$f(x, y) = \sum_i \sum_j I_{i,j} \cdot {}_{2D}h(x - i, y - j). \quad (4.3)$$

Zvyčajne symetrické a separované interpolačné jadrá sa používajú k redukovaniu výpočtovej zložitosti nasledujúcim rozdelením

$${}_{2D}h(x, y) = h(x) \cdot h(y). \quad (4.4)$$

Ilustrácia na obr. 4.1 zobrazuje interpoláciu bodu (x, y) so svojimi 4×4 susedmi. Najskôr sa interpoluje v smere osi x , kde sa vytvoria 4 medzilahlé vzorky pomocou štyroch 1D interpolácií, ktoré sú následne použité na výslednú 1D interpoláciu v smere osi y [17].



Obr. 4.1: Jednorozmerný rozklad 2D 4×4 interpolácie bodu (x, y) .

4.2 Ideálna interpolácia

Frekvenčne obmedzená funkcia, ktorá spĺňa Nyquistovo kritérium môže byť presne zrekonštruovaná použitím ideálneho rekonštrukčného filtra $\text{sinc } x$. Keďže funkcia sinc je definovaná na nekonečnom intervale, potrebujeme nekonečne veľa čísiel na zrekonštruovanie medzivzorku. Násobenie filtrom v tvare obdĺžnika vo frekvenčnej oblasti vyrežeme jedno spektrum, čo v priestorovej oblasti odpovedá konvolúcií s funkciou sinc . Funkcia sinc sa preto nazýva ideálny interpolant².

V nadväznosti na teóriu vzorkovania, rekonštrukciu 2D spojitého obrazu $f(x, y)$, je možné získať nekonečným opakovaním jeho spojitého spektra $F(u, v)$ vo Fourierovej oblasti, ktorá sa neprekrýva keďže je splnené Nyquistovo kritérium. Práve iba v tomto prípade je možné pôvodný obraz $f(x, y)$ dokonalo perfektne rekonštruovať

²Rekonštruovaná funkcia nemusí nutne prechádzať všetkými vzorkami. Používanou konvenciou je, že funkcia prechádzajúca výhradne všetkými vzorkami je *interpolant*, v ostatných prípadoch *aproximant* [9].

zo vzoriek $I_{i,j}$ vynásobením obdĺžnikovým hranolom vo Fourierovej oblasti. Ideálna 1D interpolácia je rovná násobeniu s obdĺžnikovou funkciou vo Fourierovej oblasti a realizovaná v priestorovej doméne konvolúciou so sinc funkciou

$${}^{\text{Ideal}}h(x) = \frac{\sin(\pi x)}{\pi x} = \text{sinc}(x). \quad (4.5)$$

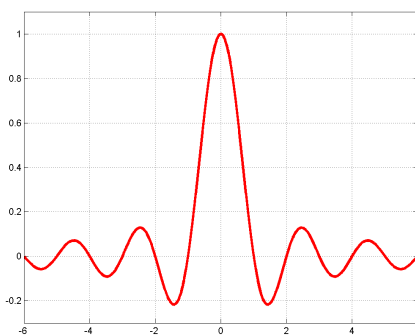
Toto jadro sa nazýva ako ideálny IIR interpolant [17]. Obraz má ale konečný počet vzoriek, preto je potrebné na rekonštrukciu použiť nasledujúce uvedené funkcie.

4.3 Sinc interpolácia

Hoci funkcia sinc poskytuje presnú rekonštrukciu signálu $f(x, y)$, jej problémom je práve zmienená priestorová neobmedzenosť. Preto sa ponúkajú dva bežné prístupy na prekonanie tohoto nedostatku. A to skrátením (*truncation*) či okienkovaním (*windowing*) funkcie za použitia okienkovej funkcie $w(x) = \text{const}(x) = 1$ a $w(x) \neq \text{const}(x)$, respektíve [17]

$${}^{\text{Sinc}}h_N(x) = \begin{cases} {}^{\text{Ideal}}h(x) \cdot w(x) & \text{pre } 0 \leq |x| < \frac{N}{2}, \\ 0 & \text{inak} \end{cases}, \quad (4.6)$$

kde N označuje počet oporných bodov konečného jadra. Najviac využívaným okien-



Obr. 4.2: Konvolyčné jadro ${}^{\text{Sinc}}h_N(x)$.

kom je tzv. Lanczos okno³, ktoré ma tvar

$$w(x) = \text{sinc}\left(\frac{x}{N}\right). \quad (4.7)$$

Dôležité je uviesť, že všetky okienkové či skrátené *sinc* jadrá sú nevyhnutne skutočnými interpolátormi [17].

³Ďalšími použitými oknami môže byť napr. Hammingovo, Kaiserovo či Dolph-Čebyševovo [21].

4.4 Interpolácia najbližším susedom (NN)

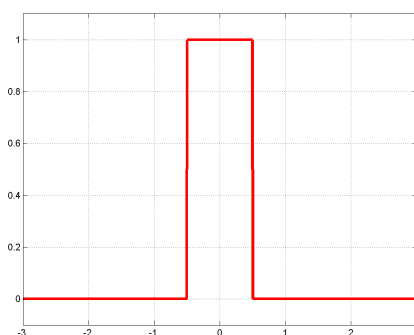
Prvou a najjednoduchšou metódou je interpolácia najbližším susedom (z ang. *Nearest Neighbour interpolation*), kedy sa ako interpolácia hodnoty v novom bode použije hodnota funkcie v najbližšom bode, kde ju poznáme. Každému interpolovanému pixelu na výstupe je priradená hodnota najbližšej vzorky bodu vo vstupnom obraze. Táto technika je tiež známa pod názvom *point shift* algoritmus. Táto najjednoduchšia interpolačná funkcia je vlastne interpoláciou prekladaním polynómu nultého stupňa. Hodnota $f(x)$ v mieste x je vybraná na základe nasledujúcej známej hodnoty I_i , preto je potrebné iba $N = 1$ podporných bodov pre určenie interpolácie. Fakticky sa to rovná konvolúcií s obdĺžnikovou funkciou. Výpočet spojitej funkcie $f(x)$ z diskkrétnej funkcie I_i definovanej v diskrétnych bodoch $i = 1, \dots, n$ má tvar [34]

$$f(x) = I_i; i - \frac{1}{2} < |x| \leq i + \frac{1}{2}, \quad (4.8)$$

a pre interpoláciu má konvolučné 1D jadro tvar [17]

$$\text{Sused} h_1(x) = \begin{cases} 1 & \text{pre } 0 \leq |x| < \frac{1}{2} \\ 0 & \text{inak} \end{cases}. \quad (4.9)$$

Princíp interpolácie spočíva v jednoduchom okopírovaní hodnoty najbližšieho suseda v okolí vzorku, implementovaná ako obyčajné zaokrúhlenie reálneho čísla. Metóda je výpočtovo značne rýchla a nový obraz obsahuje vždy iba jasové zložky obrazu pôvodného. Jej nespornou výhodou je zachovávanie práve pôvodných hodnôt jasu bodov, čím fakticky negeneruje žiadne nové dáta. Dôležité je poznamenať, že



Obr. 4.3: Konvolučné jadro $\text{Sused} h_1(x)$.

práve táto metóda interpoluje pôvodnú funkciu $f(x)$ obecné *nespojitou* funkciou, ako bolo rozobrané v úvodnej kap. 4, čo spôsobuje značný aliasing približený v kap. 2.4.5. Spomínané nežiaduce efekty skreslenia sa však prejavujú predovšetkým

na hranách s malým sklonom, kedy dochádza pri zväčšení k zreteľným skokom. A pri zmenšení poškodzuje tenké čiary, ktoré jednoducho vymiznú. Problémom je i samotné kopírovanie okolných pixelov, pretože zjavné štvorčeky farebne rovnakých susedov vyvolá ilúziu zväčšenia pixelov samotných.

4.5 Lineárna a bilineárna interpolácia

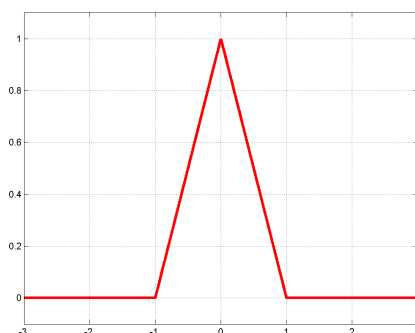
Ďalšou metódou je lineárna interpolácia, prekladanie polynómom prvého stupňa, kedy sa každá nová hodnota vypočíta ako vážený priemer dvoch známych susedných hodnôt $N = 2$ prekladaných priamkou. Pri dvoch susedných vzorkách v bodoch i a $i + 1$ s hodnotami I_i a I_{i+1} hľadáme hodnotu $f(x)$ v bode x tak, že krajnými bodmi preložíme úsečku a hľadaná hodnota v bode x sa vypočíta ako [34]

$$f(x) = I_i + \frac{x - i}{\Delta x} [I_{i+1} - I_i] ; i < |x| \leq i + 1. \quad (4.10)$$

Následne konvulučné 1D jadro odpovedajúce trojuholníkovej funkcii ma tvar [17]

$$\text{Linear } h_2(x) = \begin{cases} 1 - |x| & \text{pre } 0 \leq |x| < 1 \\ 0 & \text{inak} \end{cases}. \quad (4.11)$$

Ako bolo obecné popísané v kap. 4.1 pri digitálnom obraze je princíp prevzorovania v podstate rovnaký, kedy sa už ale nejedná o interpoláciu funkcie jednej premennej, ale dvoch. Interpoluje sa totiž súčasne v dvoch smeroch, ktorými sú



Obr. 4.4: Konvulučné jadro $\text{Linear } h_2(x)$.

riadky a stĺpce. Vzhľadom k tomu namiesto interpolácie lineárnej, kvadratickej a kubickej pojednávame už o interpolácií bilineárnej, bikvadratickej a bikubickej. Pri interpolácií farebného obrázkom v RGB, model uvedený v kap. 1.4.1, obsahujúci tri farebné kanály by bolo potrebné previesť interpoláciu u každého z kanálov. Jednalo by sa už o interpoláciu nie jednej, ale hneď troch funkcií.

Bilineárna interpolácia určuje hodnotu nového pixlu váženým priemerom 4 pixlov v najbližšom okolí 2×2 pixlu obrazu pôvodného. Povedzme, že hodnota jasovej funkcie má byť stanovená v bode o súradniciach x, y kde $\Delta x = \Delta y = 1$. Zavedme súradnice ξ, η ($0 \leq \xi, \eta < 1$) pomocou vzťahov

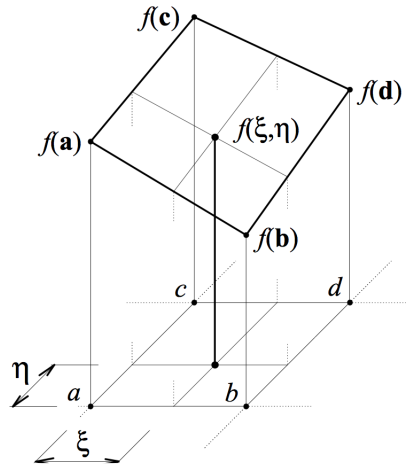
$$\xi = x - [x], \eta = y - [y]. \quad (4.12)$$

Ďalej pre stručnosť zápisu položíme

$$\begin{aligned} \mathbf{a} &= ([x], [y]), & \mathbf{b} &= ([x] + 1, [y]), \\ \mathbf{c} &= ([x], [y] + 1), & \mathbf{d} &= ([x] + 1, [y] + 1). \end{aligned} \quad (4.13)$$

Nech symboly a, b, c, d označujú body, ktoré sú reprezentované vektormi $\mathbf{a}, \mathbf{b}, \mathbf{c}, \mathbf{d}$. Bilineárna interpolácia na obr. 4.5 je popísaná nasledujúcim vzťahom

$$\begin{aligned} f(x, y) &= f([x] + \xi, [y] + \eta) \\ &= [(1 - \xi), \xi] \begin{bmatrix} f(\mathbf{a}) & f(\mathbf{b}) \\ f(\mathbf{c}) & f(\mathbf{d}) \end{bmatrix} \begin{bmatrix} (1 - \eta) \\ \eta \end{bmatrix} \\ &= (1 - \xi)(1 - \eta)f(\mathbf{a}) + \xi(1 - \eta)f(\mathbf{b}) + (1 - \xi)\eta f(\mathbf{c}) + \xi\eta f(\mathbf{d}). \end{aligned} \quad (4.14)$$



Obr. 4.5: Bilineárna interpolácia jasú. Prebrané z [28]

Hodnoty $f_x(\mathbf{a}), f_y(\mathbf{a})$ sú hodnoty derivácií (podľa x resp. podľa y) jasovej funkcie v odpovedajúcom bode. Hodnoty derivácií nahradíme diferenciami. Napr.

$$\begin{aligned} f_x(\mathbf{a}) &= \frac{f([x] + 1, [y]) - f([x] - 1, [y])}{2}, \\ f_y(\mathbf{a}) &= \frac{f([x], [y] + 1) - f([x], [y] - 1)}{2}. \end{aligned} \quad (4.15)$$

A na krajoch obrazu, kde nie sú hodnoty požadované vzťahom (4.15) k dispozícii, je možné použiť nesymetrických diferenčných vzorcov [28].

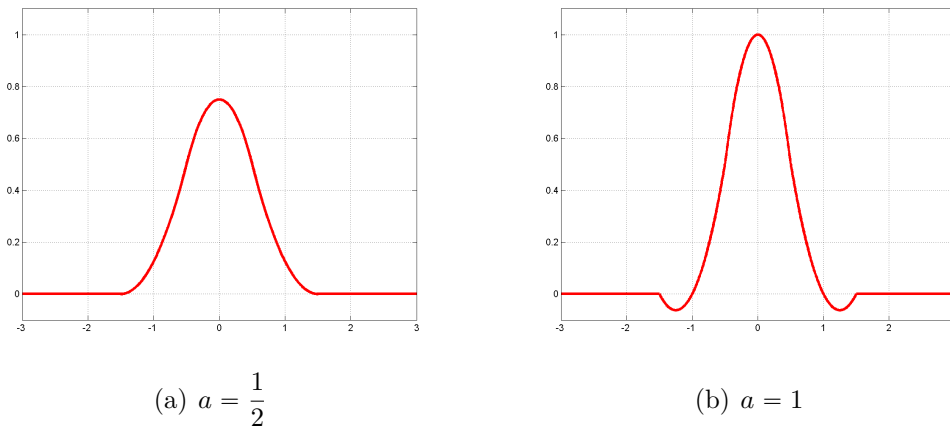
4.6 Kvadratická a bikvadratická interpolácia

Pri prekladaní 3 susedných bodov $N = 3$ pomocou kružnice či paraboly, hovoríme o kvadratickej interpolácii, kedy sa využíva polynóm druhého stupňa. Pri rekonštrukcii obrazu je kvadratická interpolácia vo veľkej miere opomínaná a v literatúre sa nato uvádzajú dva dôvody. Prvý a vážnejší namieta použitie kvadratickej interpolácie s tým, že jej filtre sú časovo premenné s fázovým skreslením [33]. Druhý subjektívny spomína, že využitie troch bodov pre interpoláciu by malo za následok dva body na jednej strane interpolovaného bodu a iba jeden bod na strane druhej. Porovnať je to možné s lineárnou rekonštrukciou, kde na každej strane je jeden bod a s kubickou rekonštrukciou, kde na každej strane sú dva body [20].

Obecné konvolučné 1D jadro má tvar [17]

$$\text{Kvadr}h_3(x) = \begin{cases} -2a|x|^2 + \frac{1}{2}(a+1) & \text{pre } 0 \leq |x| < \frac{1}{2} \\ a|x|^2 - (2a + \frac{1}{2})|x| + \frac{3}{4}(a+1) & \text{pre } \frac{1}{2} \leq |x| < \frac{3}{2} \\ 0 & \text{inak} \end{cases}, \quad (4.16)$$

kde a je parameter pre zníženie všeobecnej kvadratickej formy, $a \in \mathbb{R}$. Pre $a = \frac{1}{2}$ je jadro kvadratickým aproximátor, v prípade $a = 1$ je jadro nazývané kvadratický interpolátor [10]. Analogicky k tomu bikvadratická metóda pracuje s 9 pixlami in-



Obr. 4.6: Konvolučné jadro $\text{Kvadr}h_3(x)$.

terpolovaním 3×3 okolných susedov.

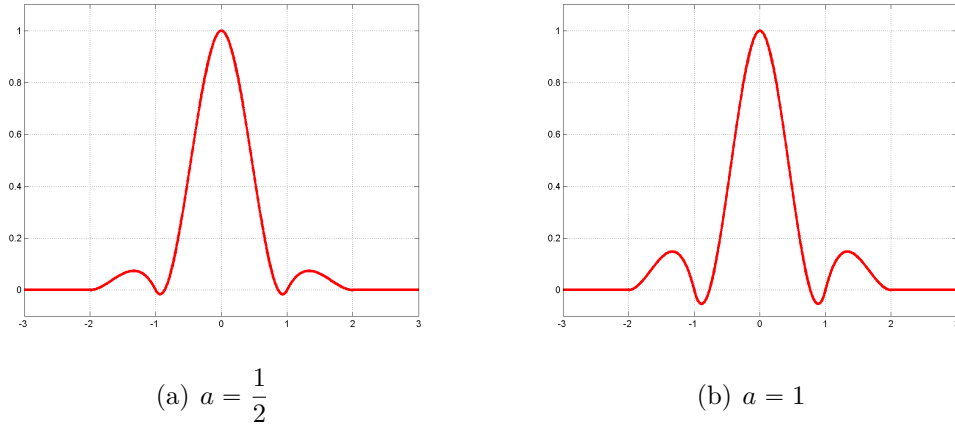
4.7 Kubická a bikubická interpolácia

Prednostnou a najpoužívanejšou metódou je kubická interpolácia, kedy sa známe hodnoty prekladajú kubickou krivkou, polynómom tretieho stupňa, vždy štyrmi su-

sednými bodmi $N = 4$. Keďže sa jedná o rôzne vzdialenosti od neznámeho pixelu, bližšie pixely zastávajú vyššiu váhu pri výpočte. Kubická interpolácia produkuje znateľne ostrejší obraz než predošlé metódy vzhľadom na prepracovanejšiu presnosť. Konvolučné jadro má obecný tvar [17]

$$\text{Kubic } h_4(x) = \begin{cases} (a+2)|x|^3 - (a+3)|x|^2 + 1 & \text{pre } 0 \leq |x| < 1 \\ a|x|^3 - 5a|x|^2 + 8a|x| - 4a & \text{pre } 1 \leq |x| < 2, \\ 0 & \text{inak} \end{cases} \quad (4.17)$$

kde a je taktiež parameter pre zníženie všeobecnej kubickej formy, $a \in \mathbb{R}$.



Obr. 4.7: Konvolučné jadro $\text{Kubic } h_4(x)$.

V tomto prípade sa bikubickou interpoláciou nový bod získava pomocou 16 bodov v najbližšom 4×4 okolí pixelu v pôvodnom obraze. Výpočet je možné previesť pomocou vzťahu

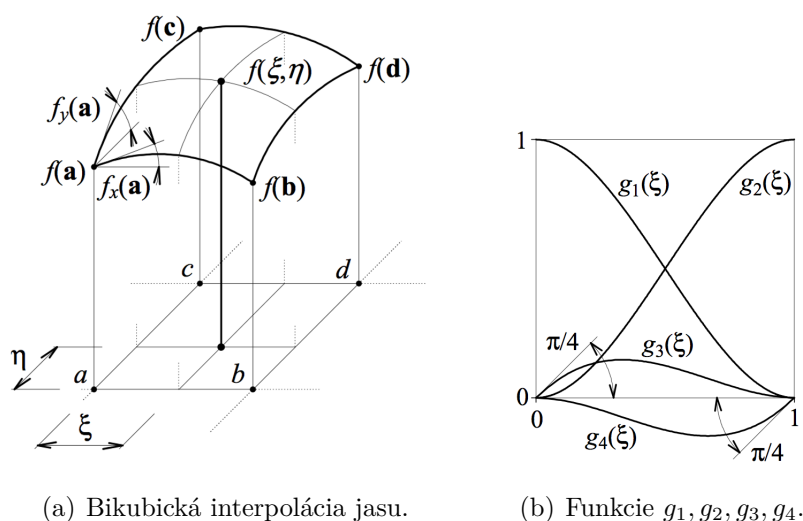
$$\begin{aligned} f(x, y) &= f(\lfloor x \rfloor + \xi, \lfloor y \rfloor + \eta) \\ &= [g_1(\xi), g_2(\xi), g_3(\xi), g_4(\xi)] \begin{bmatrix} f(\mathbf{a}) & f(\mathbf{c}) & f_y(\mathbf{a}) & f_y(\mathbf{c}) \\ f(\mathbf{b}) & f(\mathbf{d}) & f_y(\mathbf{b}) & f_y(\mathbf{d}) \\ f_x(\mathbf{a}) & f_x(\mathbf{c}) & 0 & 0 \\ f_x(\mathbf{b}) & f_x(\mathbf{d}) & 0 & 0 \end{bmatrix} \begin{bmatrix} g_1(\eta) \\ g_2(\eta) \\ g_3(\eta) \\ g_4(\eta) \end{bmatrix} \end{aligned} \quad (4.18)$$

Funkcie g_1, g_2, g_3, g_4 sú definované nasledujúcimi vzťahmi (obr. 4.8(b).)

$$\begin{aligned} g_1(\xi) &= 1 - 3\xi^2 + 2\xi, & g_2(\xi) &= (3\xi^2 - 2^3), \\ g_3(\xi) &= (\xi - 2\xi^2 + \xi^3), & g_4(\xi) &= (-\xi^2 + \xi^3). \end{aligned} \quad (4.19)$$

4.8 Fraktálna interpolácia

Tradičná geometria ponúka k interpolovaniu hodnôt v skutočnosti predovšetkým hladké tvary. Klasický spojitý obraz je väčšinou nepravidelný a jeho popis pomocou



Obr. 4.8: Bikubická interpolácia jasú a funkcie g_1, g_2, g_3, g_4 . Prebrané z [28]

hladkých kriviek býva často nepresný alebo náročný. Týmto vzniká priestor pre tzv. fraktálnu interpoláciu, ktorá prekladá známe hodnoty s určitou nepravdepodobnosťou [18]. Jedná sa už o adaptívnu metódu, kedy algoritmus pracuje v závislosti na obrazových dátach.

Pri hľadaní hodnoty v bode x preložíme fraktálom susedné vzorky i a $i - 1$ s hodnotami I_i a I_{i-1} , a získame hľadanú hodnotu funkcie $f(x)$ podľa [18]

$$f(x) = I_{i-1} + \frac{x - x_{i-1}}{x_i - x_{i-1}} [I_i - I_{i-1}] ; i - 1 < |x| \leq i. \quad (4.20)$$

Táto metóda interpolácie zahrňuje v sebe problematiku fraktálov, ktorých typickým príkladom môže byť Kochova krivka⁴. V praxi je to metóda platená ponúkaná dvoma spoločnosťami ako zásuvný modul do softvéru Photoshop. Spoločnosťami sú Altamira Group s modulom *Genuine Fractals 2.0* a spoločnosť onOne Software s modulom *Genuine Fractals 6*.

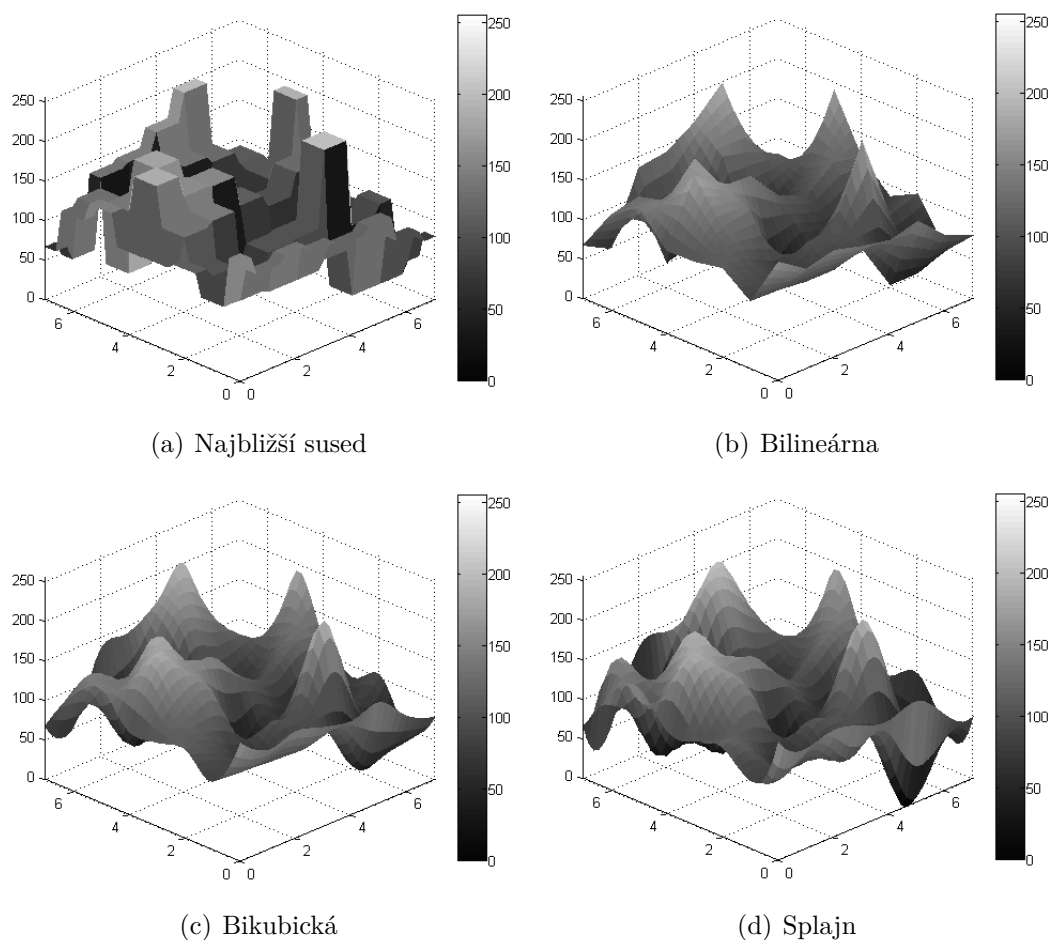
4.9 Zhrnutie

4.9.1 Rozdiely v metódach

Pre ilustráciu sú na nasledujúcom obr. 4.9 uvedené tri najbežnejšie interpolačné 2D metódy vrátane splajn rozobranej v texte neskôr. Z týchto troch najbežnejších

⁴Kochova krivka sa získava z úsečky opakovaným nahradzovaním prostrednej tretiny uniformným rovnostranným trojuholníkom, prípadne modifikovaná kochova krivka opakovaným nahradzovaním prostrednej tretiny neuniformným trojuholníkom.

metód je zrejme, že bikubická interpolácia podáva najlepšie výsledky samozrejme s daňou vysokej výpočtovej náročností. Interpolácia najbližším susedom je síce výpočtovo najmenej náročnou, avšak jej využitie je vhodné v prípade, ak je našim zámerom viditeľný raster vo výslednom obraze. Prípadne pre svoju jednoduchosť sa využíva všade tam, kde je potrebný len náhľad na digitálny obraz pri jeho spracovávaní. Typickým príkladom je nástroj lupa využívaný v počítačových softvéroch. Pre bežné využitie býva výsledný obraz príliš zubatý. Bilineárna a hlavne bikubická

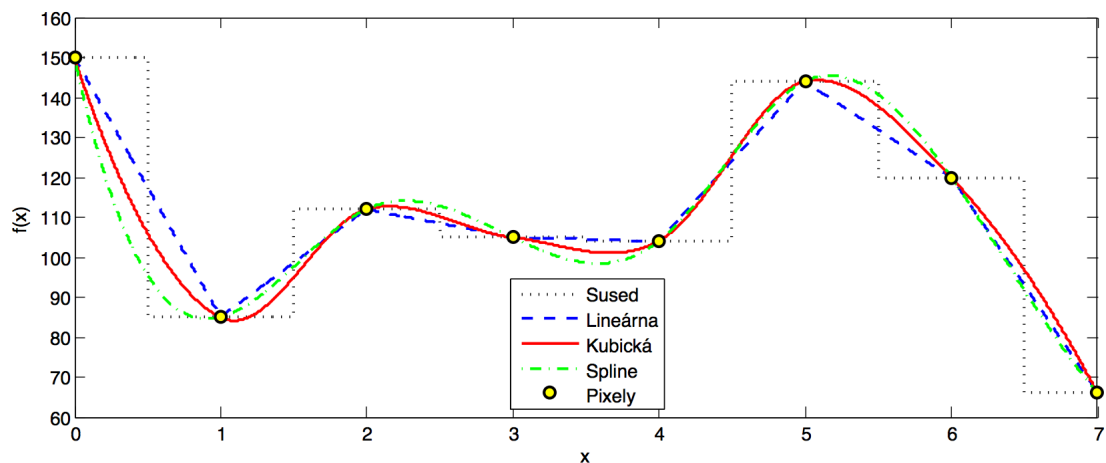


Obr. 4.9: Ilustrácia 2D interpolačných metód.

interpolácia viac vyhladzuje, čo je obecné žiadúce. Pri zmenšení obrazu spôsobuje rozostrenie, ktoré je však možné riešiť digitálnym doostrením. Niektoré zdroje doporučujú neprevádzať zmenu veľkosti obrazu skokovo, ale postupne prostredníctvom opakovaných malých zmien rozmerov. Výsledkom je zvýšený kontrast pozdĺž hrán a menšia zubatosť. Táto metóda je však veľmi prácna a navyše nespoľahlivá, pretože výsledný efekt môže byť nielen zanedbateľný, ale môže mať negatívny dopad vytvorením nových rušivých artefaktov.

4.9.2 Neadaptívne a adaptívne metódy

V stručnosti, všetky uvedené interpolácie patria medzi metódy neadaptívne, keďže interpoláciu praktikuju na všetky pixely rovnako bez rozdielu. Klasické interpolačné techniky ako najbližší sused, bilineárna či bikubická sú v značnej miere síce populárne, avšak všetky produkujú obrazy s rôznymi artefaktami (predovšetkým v okolí hrán). Najbližší sused vytvára blokové artefakty a zubatosť okrajov, zatiaľ čo bilineárna či bikubická interpolácia spôsobuje rozmazanie okrajov [6]. Tieto problémy však je možné vyriešiť práve s pomocou adaptívnych interpolačných techník. Pri adaptívnych metódach sa interpolácia mení v závislosti na celkovej štruktúre obrazu, kedy sa vyhľadávajú ostré hrany od textúry. Je to z toho dôvodu, aby bolo možné previesť interpoláciu pozdĺžne okrajov, nie cez ne. Preto tieto metódy sú schopné zredukovať artefakty a vyprodukovať obraz s ostrejšími hranami [35].



Obr. 4.10: Porovnanie jednotlivých 1D interpolácií.

5 B-SPLAJN, NURBS

Pre potrebu špeciálnych kriviek v technickej praxi nastal vývoj nových typov kriviek zadávaných riadiacimi bodmi a ďalšími parametrami. Z toho dôvodu vznikli Fergusonove (Hermitovské), Coonsove a Bézierove krivky. Následne sa objavujú B-splajn krivky a ich zovšeobecnenie nazývané NURBS krivky.

5.1 Predchodcovia B-splajn

5.1.1 Fergusonove (Hermitovské) krivky

Hermitovské krivky pomenované po francúzskom matematikovi *Charlesovi Hermitovy* patria medzi najznámejšie interpolačné krivky, často bývajú označované i ako Fergusonove krivky (sú ich zvláštnym prípadom) podľa *Jamesa C. Fergusona* (Boeing) prezentované v roku 1964 v [11]. Konkrétne sa používajú predovšetkým *Hermitovské kubiky*, tzn. polynómy 3. stupňa.

V stručnosti, tieto krivky sú tvorené dvoma riadiacimi bodmi \mathbf{P}_0 , \mathbf{P}_1 a ich dvoma dotyčnicovými vektormi \mathbf{p}'_0 a \mathbf{p}'_1 . Veľkosť týchto vektorov ovplyvňuje tvar krivky - čím je väčší, tým sa krivka k vektoru viac približuje. V prípade nulových vektorov sa z krivka stáva úsečka $\mathbf{P}_0\mathbf{P}_1$. Krivka je definovaná ako [34]

$$\text{Hermit}Q(t) = \sum_{i=0}^3 \mathbf{P}_i F_i(t), \quad t \in \langle 0, 1 \rangle. \quad (5.1)$$

Výpočet Hermitovskej kubiky $n = 3$, podľa (3.8) má tvar

$$\text{Hermit}Q(t) = \begin{bmatrix} t^3 & t^2 & t & 1 \end{bmatrix} \begin{bmatrix} 2 & -2 & 1 & 1 \\ -3 & 3 & -2 & -1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{P}_0 \\ \mathbf{P}_1 \\ \mathbf{p}'_0 \\ \mathbf{p}'_1 \end{bmatrix}, \quad (5.2)$$

následne rozpísaný do rovnice

$$\text{Hermit}Q(t) = \mathbf{P}_0 F_1(t) + \mathbf{P}_1 F_2(t) + \mathbf{p}'_0 F_3(t) + \mathbf{p}'_1 F_4(t), \quad (5.3)$$

kde F_1 , F_2 , F_3 , F_4 sú tzv. *kubické Hermitovské polynómy*¹ definované rovnicami

$$\begin{aligned} F_1(t) &= 2t^3 - 3t^2 + 1, \\ F_2(t) &= -2t^3 - 3t^2, \\ F_3(t) &= t^3 - 2t^2 + t, \\ F_4(t) &= t^3 - t^2. \end{aligned} \quad (5.4)$$

¹V odbornej lit. zaužívaná konvencia značenia bez horného indexu pre kubiku $n = 3$.

5.1.2 Bézierové krivky

Bézierove krivky poprvýkrát krát zaviedol francúzsky fyzik a matematik *Paul de Farget de Casteljau* (Citroën) v roku 1959, následne nezávisle na ňom francúzsky inžinier *Pierre Étienne Bézier* (Renault) v roku 1962. Bézierove krivky bývajú veľmi často používané v technickej praxi, typicky pri definícii písma (*font*).

Obecná Bézierová krivka n -tého stupňa je určená $n + 1$ bodmi \mathbf{P}_i riadiaceho polygónu a vzťahom [23]

$$\text{Bézier } Q(t) = \sum_{i=0}^n \mathbf{P}_i B_i^n(t), \quad t \in \langle 0, 1 \rangle. \quad (5.5)$$

kde B_i^n sú *Bernsteinove polynómy* n -tého stupňa

$$B_i^n = \binom{n}{i} t^i (1-t)^{n-i}; \quad t \in \langle 0, 1 \rangle; \quad i = 0, 1, \dots, n, \quad (5.6)$$

kde $\binom{n}{0} = 1, 0^0 = 1$.

Typicky používanými krivkami sú predovšetkým Bézierové kubiky $n = 3$, kde $\mathbf{P}_0, \mathbf{P}_1, \mathbf{P}_2, \mathbf{P}_3$ sú štyri riadiace body. Krivka vychádza z prvého a končí v poslednom riadiacom bode, analogicky k (5.5) je určená vzťahom

$$\text{Bézier } Q(t) = \sum_{i=0}^3 \mathbf{P}_i B_i(t), \quad t \in \langle 0, 1 \rangle. \quad (5.7)$$

Výpočet Bézierovej kubiky $n = 3$, podľa (3.8) má tvar

$$\text{Bézier } Q(t) = \begin{bmatrix} t^3 & t^2 & t & 1 \end{bmatrix} \begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 3 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{P}_0 \\ \mathbf{P}_1 \\ \mathbf{P}_2 \\ \mathbf{P}_3 \end{bmatrix}, \quad (5.8)$$

kde $\mathbf{P}_0, \mathbf{P}_1, \mathbf{P}_2, \mathbf{P}_3$ sú štyri riadiace body, ktoré je možno chápať ako maticu o veľkosti 4×2 . Následne matica rozpísaná do rovnice má tvar

$$\text{Bézier } Q(t) = \mathbf{P}_0 B_0(t) + \mathbf{P}_1 B_1(t) + \mathbf{P}_2 B_2(t) + \mathbf{P}_3 B_3(t), \quad (5.9)$$

kde *kubické Bernsteinove polynómy*² majú tvar

$$\begin{aligned} B_0(t) &= (1-t)^3, \\ B_1(t) &= 3t(1-t)^2, \\ B_2(t) &= 3t^2(1-t), \\ B_3(t) &= t^3. \end{aligned} \quad (5.10)$$

Dotyčnicové vektory v prvom a poslednom bode majú tvar

$$\mathbf{p}'(0) = 3(\mathbf{P}_1 - \mathbf{P}_0), \quad \mathbf{p}'(1) = 3(\mathbf{P}_3 - \mathbf{P}_2). \quad (5.11)$$

²V odbornej lit. zaužívaná konvencia značenia bez horného indexu pre kubiku $n = 3$.

Vlastnosti Bernsteinových polynómov

Bernsteinove polynómy B_i^n zo vzťahu (5.6) majú nasledujúce vybrané vlastnosti [34, 26]:

- Nezápornosť Bernsteinových polynómov:

$$\forall i, n \in \mathbb{N} \cup \{0\} \quad a \quad B_i^n(t) \geq 0 \quad \text{pre } t \in \langle 0, 1 \rangle.$$

- Súčet Bernsteinových polynómov je rovný jednej (*partition of unity*):

$$\sum_{i=0}^n B_i^n(t) = 1 \quad \text{pre } t \in \langle 0, 1 \rangle.$$

Spolu s predchádzajúcim vzťahom zaručuje, že výsledná krivka bude vždy ležať v konvexnej obálke bodov riadiaceho polygónu.

- $B_i^n(t)$ majú iba jedno maximum na intervale $t \in \langle 0, 1 \rangle$, a to v čase $t = i/n$.
- Pre koncové body platí:

$$B_i^n(0) = \begin{cases} 1 & \text{pre } i = 0 \\ 0 & \text{inak,} \end{cases} \quad B_i^n(1) = \begin{cases} 1 & \text{pre } i = n \\ 0 & \text{inak.} \end{cases}$$

- Plocha pod všetkými polynómami je rovnaká:

$$\int_0^1 B_i^n(t) dt = \frac{1}{n+1}.$$

- Pre rekurentnú definíciu Bernsteinova polynómu stupňa n pomocou lineárnej kombinácie dvoch po sebe nasledujúcich Bernsteinových polynómov stupňa $n-1$ platí:

$$B_i^n(t) = (1-t)B_i^{n-1}(t) + tB_{i-1}^{n-1}(t).$$

Vlastnosti Bézierových kriviek

Bézierové krivky zo vzťahu (5.5) majú nasledujúce vybrané vlastnosti [34, 26, 24]:

- Dosadením $t = 0$, resp. $t = 1$ do vzťahu (5.5) sa získa Bézierová krivka prechádzajúca koncovými bodmi \mathbf{P}_0 a \mathbf{P}_n :

$$\text{Bézier } Q(0) = \mathbf{P}_0, \quad \text{Bézier } Q(1) = \mathbf{P}_n.$$

Čiže nejedná sa ani o aproximačnú, ani o interpolačnú krivku.

- Dosadením $t = 0$, resp. $t = 1$ do derivácie vzťahu (5.5) sa získajú výrazy pre dotyčnicové vektory v krajných bodoch:

$$\mathbf{p}'_0 = n(\mathbf{P}_1 - \mathbf{P}_0), \quad \mathbf{p}'_1 = n(\mathbf{P}_3 - \mathbf{P}_2). \quad (5.12)$$

- Zmenou polohy jedného riadiaceho bodu \mathbf{P}_i dochádza ku zmene tvaru celej krivky.
- Bézierové krivky sú invariantné voči afinným transformáciám.

5.1.3 Coonsove krivky

Coonsove krivky, ktoré sú špeciálnou formou B-splajn kriviek zaviedol *Steven Anson Coonsom* (MIT, Ford) v roku 1964. Jediným rozdielom je tvar bázových polynómov. V technickej praxi sa stále jedna o krivky, ktoré sú v značnej miere používané, konkrétne Coonsove kubiky $n = 3$. Coonsova kubika je znova určená štyrmi riadiacimi bodmi $\mathbf{P}_0, \mathbf{P}_1, \mathbf{P}_2, \mathbf{P}_3$. Krivka narozdiel od predošlých nezačína ani nekončí v žiadnom z týchto bodov a je určená vzťahom

$${}^{\text{Coons}}Q(t) = \frac{1}{6} \sum_{i=0}^3 \mathbf{P}_i C_i(t), \quad t \in \langle 0, 1 \rangle. \quad (5.13)$$

Výpočet Coonsovej kubiky $n = 3$, podľa (3.8) má maticový tvar

$${}^{\text{Coons}}Q(t) = \frac{1}{6} \begin{bmatrix} t^3 & t^2 & t & 1 \end{bmatrix} \begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 0 & 3 & 0 \\ 1 & 4 & 1 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{P}_0 \\ \mathbf{P}_1 \\ \mathbf{P}_2 \\ \mathbf{P}_3 \end{bmatrix}, \quad (5.14)$$

následne rozpísaný do rovnice

$${}^{\text{Coons}}Q(t) = \mathbf{P}_0 C_0(t) + \mathbf{P}_1 C_1(t) + \mathbf{P}_2 C_2(t) + \mathbf{P}_3 C_3(t), \quad (5.15)$$

kde bázove *kubické Coonsove polynómy*³ majú tvar

$$\begin{aligned} C_0(t) &= \frac{(1-t)^3}{6}, \\ C_1(t) &= \frac{3t^3 - 6t^2 + 4}{6}, \\ C_2(t) &= \frac{-3t^3 + 3t^2 + 3t + 1}{6}, \\ C_3(t) &= \frac{t^3}{6}. \end{aligned} \quad (5.16)$$

Krivka obecné neprechádza krajnými bodmi svojho riadiaceho polygónu, začína a končí v bodoch

$$Q(0) = \frac{\mathbf{P}_0 + 4\mathbf{P}_1 + \mathbf{P}_2}{6}, \quad Q(1) = \frac{\mathbf{P}_1 + 4\mathbf{P}_2 + \mathbf{P}_3}{6}, \quad (5.17)$$

ktoré ležia v tzv. antifažiskách trojuholníkov $\mathbf{P}_0\mathbf{P}_1\mathbf{P}_2$ a $\mathbf{P}_1\mathbf{P}_2\mathbf{P}_3$. Dotyčnicové vektory v prvom a poslednom bode majú tvar

$$\mathbf{p}'(0) = \frac{\mathbf{P}_2 - \mathbf{P}_0}{2}, \quad \mathbf{p}'(1) = \frac{\mathbf{P}_3 - \mathbf{P}_1}{2}. \quad (5.18)$$

³V odbornej lit. zaužívaná konvencia značenia bez horného indexu pre kubiku $n = 3$.

Jednotlivé Coonsove kubiky je možné hladko napájať, čím vzniká po častiach polynomickeá krivka, ktorá sa nazýva *Coonsov kubický B-splajn*, prípadne uniformný neracionálny kubický B-splajn. Body styku sa nazývajú uzly a bývajú označené t . Coonsov B-splajn je určený $n \geq 4$ bodmi a je tvorený $n - 3$ segmentami. Parameter t definuje tzv. uzlový vektor. Zovšeobecnením Coonsovej kubickej B-splajn krivky sa získa B-splajn s ľubovoľným stupňom.

5.2 B-splajn, NURBS krivky

5.2.1 B-splajn bázové funkcie

Ako bázové funkcie sa pre B-splajn a NURBS používajú B-splajn funkcie definované rekurentne [25]. Nech $\mathbf{t} = (t_0, t_1, \dots, t_m)$ je uzlový vektor. B-splajn funkcia je definovaná ako

$$N_i^0(t) = \begin{cases} 1 & \text{pre } t \in \langle t_i, t_{i+1} \rangle \\ 0 & \text{inak,} \end{cases}$$

$$N_i^n(t) = \frac{t - t_i}{t_{i+n} - t_i} N_i^{n-1}(t) + \frac{t_{i+n+1} - t}{t_{i+n+1} - t_{i+1}} N_{i+1}^{n-1}(t), \quad (5.19)$$

kde $0 \leq i \leq m - n - 1$, $1 \leq n \leq m - 1$, $\frac{0}{0} := 0$.

Uzlový vektor je tvorený neklesajúcou postupnosťou kladných reálnych čísiel. Obvykle sa zadáva v intervale $\langle 0, 1 \rangle$, obecné však je možné voliť ľubovoľný interval. Pokiaľ je priebeh plynulý, jedná sa o *ekvidistantný* (uniformný) uzlový vektor, v opačnom prípade sa nazýva *neekvidistantný* (neuniformný). Uzlový vektor $\mathbf{t} = (t_0, t_1, \dots, t_m)$, ktorého počet prvkov $m + 1$ nazývame *dĺžka uzlového vektoru*, je ekvidistantný ak platí

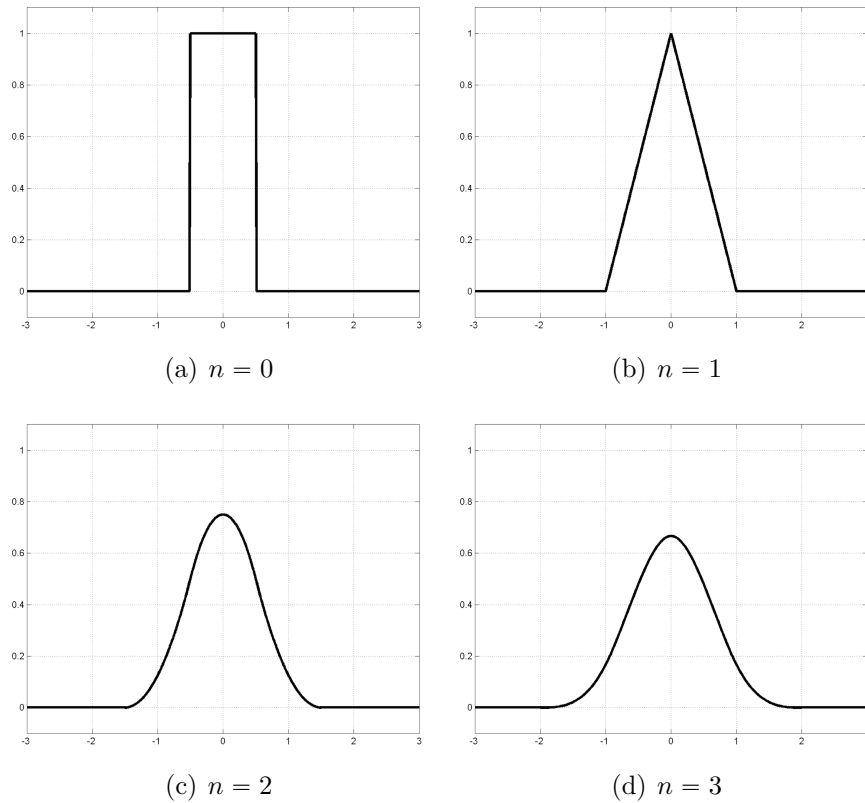
$$t_{i+1} - t_i = t_{i+2} - t_{i+1}, \quad \text{kde } i = 0, 1, \dots, t_{m-2}. \quad (5.20)$$

Pokiaľ nie je rovnosť (5.20) splnená, jedná sa o uzlový vektor neekvidistantný [23].

5.2.2 B-splajn krivky

B-splajny (z ang. *Basic Spline*) sú jednými z najviac využívaných z rodiny splajnov, ktoré sa skladajú zo segmentov Bézierových kriviek⁴ rovnakého stupňa splňujúce podmienku C^2 spojitosti. Zaujímavosťou je, že u B-splajn kriviek prebieha spojitý napájanie kriviek automaticky. Výhodou je, že v závislosti na voľbe stupňa B-splajn krivky a dĺžke riadiaceho polygónu môžu B-splajn krivky omnoho viac kopírovať riadiaci polygón. Odstraňujú najväčšie nevýhody Bézierových kriviek a to, že zmenou

⁴B-splajn krivky sú zovšeobecnením Bézierových kriviek.



Obr. 5.1: B-splajn funkcie stupňa $n = 0, 1, 2, 3$ centrovane pre $t = 0$.

polohy riadiaceho bodu je možné krivku meniť iba lokálne. A následne s rastúcim počtom bodov riadiaceho polygónu nerastie nutne stupeň krivky, čím sa krivka nevzdalaže od svojho riadiaceho polygónu.

Nech je dané $m + 1$ riadiacich kontrolných bodov $\mathbf{P}_0, \mathbf{P}_1, \dots, \mathbf{P}_m$ kde $\mathbf{P}_i \in \mathbb{R}$, uzlový vektor $\mathbf{t} = (t_0, t_1, \dots, t_{m+n+1})$. B-spline krivka stupňa n pre riadiace body \mathbf{P}_i a uzlový vektor \mathbf{t} je definovaná ako

$$\text{B-splajn } Q(t) = \sum_{i=0}^m \mathbf{P}_i N_i^n(t), \quad (5.21)$$

kde N_i^n sú bázové B-spline funkcie podľa definície (5.19) v predchádzajúcej časti 5.2.1. Riadiace body $\mathbf{P}_0, \mathbf{P}_1, \dots, \mathbf{P}_m$ sa nazývajú *deBoorove body* a ich spojenie sa nazýva *deBoorov polygonálny ťah*. Zmena kontrolného bodu \mathbf{P}_i ovplyvňuje krivku iba v časti odpovedajúcej intervalu $\langle t_i, t_{i+n+1} \rangle$ [23]. Dĺžka uzlového vektoru sa musí rovnať súčtu počtu bodov a stupňa krivky plus jedna.

5.2.3 NURBS krivky

Neuniformné racionálne B-splajn krivky (NURBS z ang. *Non-Uniform Rational Basis Spline*) predstavujú dvoje zovšeobecnenie uniformnej B-splajn reprezentácie.

NURBS krivka rovnako ako B-splajn je konštruovaná ako lineárna kombinácia básových funkcií, kde súčinitelia u jednotlivých básových funkcií sú tzv. riadiace body. Kvalita napojenia jednotlivých segmentov splňuje podmienku C^{n-1} spojitosti, kde n je stupeň krivky. Pri rovnakých rozdieloch hodnôt parametru, v ktorých dochádza k napojeniu jednotlivých segmentov kriviek sa jedna o krivky uniformné [23, 22].

Prvé zovšeobecnenie uniformnej B-splajn reprezentácie je neuniformná B-splajn reprezentácia, kedy rozdiely hodnôt parametru, v ktorých dochádza k napojeniu jednotlivých segmentov krivky, nie sú rovnaké. Druhým zovšeobením je racionalita, ktorá narozdiel od polynomickej reprezentácie priraduje každému riadiacemu bodu váhu. Táto váha následne určuje s akou silou bude bod pôsobiť na tvar krivky. Vďaka tomuto zovšeobecneniu je už možné modelovať kuželosečkové oblúky (eliptický, parabolický a hyperbolický).

Majme daných $m + 1$ kontrolných bodov \mathbf{P}_i , $m + 1$ kladných reálnych čísiel w_i nazývaných váhy, ďalej stupeň krivky n a uzlový vektor $\mathbf{t} = (t_0, t_1, \dots, t_{m+n+1})$. NURBS krivka je definovaná ako

$${}^{\text{NURBS}}Q(t) = \frac{\sum_{i=0}^m w_i \mathbf{P}_i N_i^n(t)}{\sum_{i=0}^m w_i N_i^n(t)}, \quad (5.22)$$

kde $t \in \langle t_n, t_{m+1} \rangle$. Ak označíme

$$R_i^n(t) = \frac{w_i N_i^n(t)}{\sum_{i=0}^m w_i N_i^n(t)}, \quad (5.23)$$

potom je možné NURBS krivku zaposvať ako

$${}^{\text{NURBS}}Q(t) = \sum_{i=0}^m \mathbf{P}_i R_i^n(t). \quad (5.24)$$

5.2.4 Problém a riešenie interpolácie NURBS

Väčšina aplikácií NURBS kriviek obecného tvaru je založená na použití a konštrukcií kriviek aproximačných. Tento typ je vhodný a žiadaný pri dátach, ktoré sú zaťažené nejakou chybou. V prípade, že sa pracuje s presnými dátami, nie je účelné navrhovať aproximačnú krivku, ktorá zadanými riadiacimi bodmi neprechádza. Takto je nutné modelovať interpolačnú krivku, ktorá zadanými tzv. *definičnými* bodmi prechádza.

Toto nastoľuje hlavný problém, že interpolačnú krivku nie je možné počítať priamo. Síce je známa presná poloha definičných bodov, ktoré ležia na krivke, ale nie sú známe polohy riadiacich bodov, ktoré vystupujú v rovniciach pre NURBS krivky. Preto je potrebné najskôr zostaviť sústavu rovníc, ktorej riešením sú súradnice neznámych riadiacich bodov, a až potom je možné pristúpiť k návrhu interpolačnej krivky.

K problému interpolácie je možné rozlíšiť dva základné prístupy. V prvom prípade sa vychádza z podmienky, že počet neznámych riadiacich bodov ma byť rovnaký ako je počet definičných bodov [23]. Tento spôsob interpolácie je označovaný jednoduchá interpolácia a výsledná krivka je jednoduchou interpolačnou krivkou. Druhý prípad vychádza z logického požiadavku, aby výsledná interpolačná krivka mala práve toľko segmentov, koľko je ramien riadiaceho polygónu, a aby uzly oddeľujúce jednotlivé segmenty ležali priamo v definičných bodoch [16]. Tento spôsob je označovaný ako uzlová interpolácia a výslednou interpolačnou krivkou je uzlová interpolačná krivka. Počet riadiacich bodov je v tom prípade väčší než počet bodov definičných, preto je nutné pri návrhu uzlovej interpolačnej krivky stanoviť ďalšie (spravidla okrajové) podmienky [23].

Okrem samotnej polohy definičných bodov ovplyvňuje výsledný tvar jednoduchej interpolačnej krivky stupeň, vektor parametrizácie, uzlový vektor a váhy riadiacich bodov. Pri uzlovej interpolačnej krivky je daný polohou definičných bodov, stupňom, vektorom parametrizácie, váhami riadiacich bodov a okrajovými podmienkami.

Nech je predpokladané, že je dostupná množina $(m + 1)$ nameraných hodnôt

$$\mathbf{Q}_k, \quad k = 0, \dots, m. \quad (5.25)$$

Na zostrojenie neuniformnej B-splajn krivky stupňa n , ktorá bude body interpolovať je daný vzťah

$$\mathbf{Q}_k = {}^{\text{NURBS}}Q(\bar{t}_k) = \sum_{i=0}^m \mathbf{P}_i R_i^n(\bar{t}_k). \quad (5.26)$$

Najskôr je potrebné spočítať vhodné hodnoty \bar{t}_k . K ich určeniu je možné použiť tetivovú alebo stredovú metódu. Nech d značí absolútnu dĺžku.

Tetivová metóda:

$$d = \sum_{k=1}^m |\mathbf{Q}_k - \mathbf{Q}_{k-1}|. \quad (5.27)$$

Potom

$$\begin{aligned} \bar{t}_0 &= 0 \quad \text{a} \quad \bar{t}_m = 1, \\ \bar{t}_k &= \bar{t}_{k-1} + \frac{|Q_k - Q_{k-1}|}{d}, \quad k = 1, \dots, m-1. \end{aligned} \quad (5.28)$$

Stredová metóda:

$$d = \sum_{k=1}^m \sqrt{|\mathbf{Q}_k - \mathbf{Q}_{k-1}|}. \quad (5.29)$$

Potom

$$\begin{aligned} \bar{t}_0 &= 0 \quad \text{a} \quad \bar{t}_m = 1, \\ \bar{t}_k &= \bar{t}_{k-1} + \frac{\sqrt{|Q_k - Q_{k-1}|}}{d}, \quad k = 1, \dots, m-1. \end{aligned} \quad (5.30)$$

Pomocou tetivovej alebo stredovej metódy sa spočítajú parametry $\bar{t}_0, \dots, \bar{t}_m$. Pre výpočet riešiteľnej sústavy pre rovnicu (5.26) je nutné previesť metódu nazývanú

priemerovanie (z ang. *averaging*) [23]:

$$t_0 = \dots = t_n = 0 \quad \text{a} \quad t_{m-n} = \dots = 1,$$

$$t_{j+n} = \frac{1}{n} \sum_{i=j}^{j+n-1} \bar{t}_i, \quad j = 1, \dots, m-n. \quad (5.31)$$

Po výpočte uzlových vektorov zostáva určiť riadiace body. Rovnica (5.26) určuje $(m+1) \times (m+1)$ lineárnych rovníc neznámych \mathbf{P}_i . Jej prvky sa dostanú výpočtom bázových funkcií s parametrami $\bar{\mathbf{t}}$ pre uzlový vektor \mathbf{t} .

$$\begin{bmatrix} R_0^n(\bar{t}_0) & R_1^n(\bar{t}_0) & \dots & R_m^n(\bar{t}_0) \\ R_0^n(\bar{t}_1) & R_1^n(\bar{t}_1) & \dots & R_m^n(\bar{t}_1) \\ \vdots & \vdots & \ddots & \vdots \\ R_0^n(\bar{t}_m) & R_1^n(\bar{t}_m) & \dots & R_m^n(\bar{t}_m) \end{bmatrix} \begin{bmatrix} \mathbf{P}_0 \\ \mathbf{P}_1 \\ \vdots \\ \mathbf{P}_m \end{bmatrix} = \begin{bmatrix} \mathbf{Q}_0 \\ \mathbf{Q}_1 \\ \vdots \\ \mathbf{Q}_m \end{bmatrix}. \quad (5.32)$$

Tvar interpolačnej krivky môže byť ovplyvnený i váhami riadiacich bodov, ktoré majú však iný význam než váhy riadiacich bodov aproximačných kriviek. U aproximačnej krivky má vyšší stupeň váha riadiaceho bodu za následok priblíženie krivky k tomuto riadiacemu bodu. V prípade interpolačnej krivky, kde je poloha riadiaceho bodu neznáma, je potrebné efekt váh chápať opačne. Zvýšením váhy riadiaceho bodu interpolačnej krivky sa približuje riadiaci bod viac k definičnému polygónu. To vedie vo svojom dôsledku k plynulejšiemu priebehu interpolačnej krivky a vhodne stanovené váhy obmedzia nežiadúce prekmity a zvlnenie výslednej krivky. Neznámym riadiacim bodom sa preto vopred priradia vhodné váhy pre určenie racionálnych bázových funkcií a zostaví sa sústava rovníc pre výpočet riadiacich bodov. Dodatočnou zmenou váh by sa menila príslušná racionálna bázová funkcia čo spôsobí, že interpolačná krivka už neprechádzala definičnými bodmi [16].

Ďalšie interpolačné metódy

Pre doplnenie existujú ešte nasledujúce metódy interpolácie. Medzi ďalšie interpolačné metódy teda patrí napríklad Lagrangerova interpolácia, využívajúca Lagrangerove polynómy viac v [17, 5]. Následne interpolácie využívajú interpolačné kubiky odvodené od Hermitovských polynómov ako Akimova interpolácia rozobraná v [1], Kochanek-Bartles splajn (TCB Splajny) viac v [7], špeciálny Catmull-Rom splajn [15], Čebyšev-Gauss kvadratura priblížená v [14] a iné.

6 METÓDY HODNOTENIA PODOBNOSTI OBRAZU

Pri dosiahnutí výsledných rekonštruovaných obrazov je potrebné zhodnotiť ich kvalitu a prevedenie jednotlivými interpolačnými metódami. Nato slúžia rôzne objektívne a subjektívne metódy hodnotenia obrazu, opisujúce podanie kvalitatívnej vernosti, či mieru skreslenia medzi signálom pôvodným a signálom skresleným.

Objektívne metódy hodnotia obraz na základe explicitných numerických kritérií oproti subjektívnym, ktoré sú založené výhradne na ľudskom úsudku. Tieto metódy majú za úlohu porovnávať transformovaný obraz z hľadiska kvality po jeho rekonštrukcii interpolačnými metódami. V rámci textu sú rozobrané porovnávacie metódy objektívne, akými sú *stredná kvadratická chyba* (MSE), *odstup signálu od šumu* (SNR), *špičkový odstup signálu od šumu* (PSNR) a *index štruktúrálnej odobnosti* (SSIM). Keďže výhradne objektívne metódy nedokážu napodobniť vnímanie obrazu človekom, následne sú priblížené i subjektívne metódy vychádzajúce z doporučenia ITU-R BT.500-11.

6.1 Objektívne metódy hodnotenia

6.1.1 Stredná kvadratická chyba (MSE)

Stredná kvadratická chyba (z ang. *Mean Square Error*) je najjednoduchšou a najpoužívanejšou metódou pre hodnotenie digitálneho obrazu. Vyjadruje súčet kvadrátov odchýlok v jednotlivých pixeloch medzi pôvodným a transformovaným obrazom. Pre 2D obrazový signál v stupňoch šedi je definovaná nasledujúcou rovnicou

$$\text{MSE} = \frac{1}{MN} \sum_{i=1}^M \sum_{j=1}^N (I_{i,j} - I'_{i,j})^2, \quad (6.1)$$

kde $I_{i,j}$ je pôvodným obrazom a $I'_{i,j}$ hodnoteným obrazom o veľkosti $M \times N$. MSE určuje vizuálnu kvalitu obrazového signálu udávajúc rozdielový chybový signál, daný výrazom $I_{i,j} - I'_{i,j}$, medzi pôvodným a transformovaným obrazom [30].

6.1.2 Odstup signálu od šumu (SNR)

Odstup signálu od šumu (z ang. *Peak Signal to Noise Ratio*) udáva pomer medzi energiou obrazového signálu k energii šumu. Pre 2D obrazový signál v stupňoch šedi

je definovaná nasledujúcou rovnicou

$$\text{SNR} = 10 \log_{10} \left(\frac{\sum_{i=1}^M \sum_{j=1}^N (I_{i,j})^2}{\sum_{i=1}^M \sum_{j=1}^N (I_{i,j} - I'_{i,j})^2} \right), \quad (6.2)$$

kde $I_{i,j}$ je pôvodným obrazom a $I'_{i,j}$ hodnoteným obrazom o veľkosti $M \times N$. SNR je vyjadrovaná v decibeloch (dB).

6.1.3 Špičkový odstup signálu od šumu (PSNR)

Špičkový odstup signálu od šumu (z ang. *Peak Signal to Noise Ratio*) vychádzajúci z výpočtu hodnoty MSE sa zavádza z dôvodu, že porovnávané obrazy môžu mať rozdielne dynamické rozsahy. Udáva pomer medzi maximálnou možnou energiou obrazového signálu k energii šumu, pričom pri hodnote MSE rovnej nule dosahuje do nekonečna. Štandardný PSNR pomer je definovaný ako

$$\text{PSNR} = 10 \log_{10} \left(\frac{L^2}{\text{MSE}} \right) = 20 \log_{10} \left(\frac{L}{\sqrt{\text{MSE}}} \right), \quad (6.3)$$

kde L je konštanta, reprezentujúca maximálny dynamický rozsah intenzity pixlu v obraze (napr. pre 8 bitov/pixel v stupňoch šedi, $L = 2^8 - 1 = 255$). PSNR je vyjadrovaná v decibeloch (dB), kedy bežne dosahuje hodnoty v rozmedzí 20–40 dB [30].

6.1.4 Štruktúrálna podobnosť (SSIM)

Štruktúrálna podobnosť (z ang. *Structural Similarity*) je ďalšia metóda vyjadrujúca podobnosť dvoch obrazov. Berie na vedomie skutočnosť, že ľudské videnie či vnímanie scény je vysoko prispôbené k extrahovaniu štruktúrálnej informácie. SSIM bola navrhnutá s cieľom vylepšiť tradičné metódy hodnotenia kvality obrazu ako PSNR či MSE [31, 32].

Hlavnou vlastnosťou je jej prispôbenie sa ľudskému systému videnia (HVS), ktorý je rozobraný v kap. 1.3. SSIM nesčítava chyby ako predchádzajúce metódy, ale obrazové skreslenie definuje pomocou kombinácie troch faktorov. Týmito faktormi sú *strata korelácie*, *skreslenie svietivosti* a *skreslenie kontrastu*. Definovaná je ako

$$\text{SSIM} = l(x, y), c(x, y), s(x, y) \begin{cases} l(x, y) = \frac{2\mu_x\mu_y + C_1}{\mu_x^2\mu_y^2 + C_1} \\ c(x, y) = \frac{2\sigma_x\sigma_y + C_2}{\sigma_x^2\sigma_y^2 + C_2} \\ s(x, y) = \frac{\sigma_{xy} + C_3}{\sigma_x\sigma_y + C_3} \end{cases}, \quad (6.4)$$

kde $l(x, y)$ je porovnávací funkcia svietivosti dvoch obrazov k strednej svietivosti μ_x a μ_y s faktorom maximálne rovný 1 za podmienky $\mu_x = \mu_y$. Funkcia $c(x, y)$ porovnáva kontrast dvoch obrazov za pomoci odchýlok $\sigma_x \sigma_y$, ktorá je taktiež maximálne rovná 1 za podmienky $\sigma_x = \sigma_y$. Nakoniec $s(x, y)$ porovnáva štruktúry meraním korelačným koeficientom medzi obrazmi x a y . A zavedené konštanty C_1, C_2, C_3 s kladnou hodnotou sú použité za účelom nenulového menovateľa.

SSIM nadobúda hodnoty od 0 do 1, kde dolná hranica 0 značí nulovú koreláciu medzi obrazmi a horná hranica 1 značí totožnosť obrazov.

6.2 Subjektívne metódy hodnotenia

6.2.1 Dvojitá stimulovaná miera zhoršenia (DSIS)

Dvojitá stimulovaná miera zhoršenia (z ang. *Double Stimulus Impairment Scale*) je metóda hodnotenia skreslenia obrazu, pri ktorej je k dispozícii neskreslená predloha reprezentujúca plnú kvalitu. Predpokladom tejto metódy je, že oproti referenčnému snímku bude mať skreslený obraz rovnakú či horšiu kvalitu. Pokiaľ by tento predpoklad bol opačný, navrhnutá stupnica by fakticky nedostačovala a dôjde k strate drahocennej informácie. Pri tomto hodnotení pozorovateľ pozná už vopred referenčný obraz a je dotazovaný iba na hodnotenie skresleného obrazu. V prevažnej väčšine sa využíva 5stupňová škála, viz [27].

6.2.2 Jednoduchá stimulácia (SS)

Jednoduchá stimulácia (z ang. *Single Stimulus*) je metóda hodnotenia skreslenia obrazu, pri ktorej nie je k dispozícii neskreslená predloha reprezentujúca plnú kvalitu. Obrázok sa ukazuje na krátku dobu (3-5 sekúnd) a následne sa volí kvalita obrazu na stupnici. Testovaný obrázok sa strieda so šedou farbou. V prevažnej väčšine sa využíva 5stupňová škála stupnice, viz [27].

7 POROVNANIE METÓD V PROGRAME MATLAB[®]

V tejto kapitole sú uvedené programové riešenia bežných i pokročilých spôsobov interpolácie. Prvú časť tvoria popísané interpolačné metódy implementované v programovom prostredí MATLAB[®]. Interpolácia prebiehala na pôvodných obrázkoch v stupňoch šedi (jasové intenzity v rozsahu 0 až 255) o veľkosti 512×512 pixelov, ktoré pred interpoláciou boli zmenšené funkciou `imresize` s merítkom $scale = 4$ na veľkosť 128×128 pixelov a následne interpolované s rovnakou hodnotou merítka $scale = 4$ na pôvodných 512×512 pixelov. Tento prístup k interpolácii bol následne použitý pre všetky naprogramované funkcie interpolácie. Pre tento spôsob prístupu bolo možné následne vypočítať hodnoty pre objektívne metódy hodnotenia:

- stredná kvadratická chyba (MSE)
- odstup signálu od šumu (SNR)
- odstup špičkového signálu od šumu (PSNR)
- štruktúrna podobnosť (SSIM)

Testované obrázky spolu s odpovedajúcimi hodnotami objektívnych metód sú generované v prílohe. Pre subjektívne metódy hodnotenia bola použitá metóda dvojnásobne stimulovaného merítka kvality (DSIS). Konkrétne sa jednalo iba o 4 druhy obrázkov a to *lena.png*, *barbara.png*, *zoneplate.png*, *flinstones.png*.

7.1 Programové riešenie interpolácie

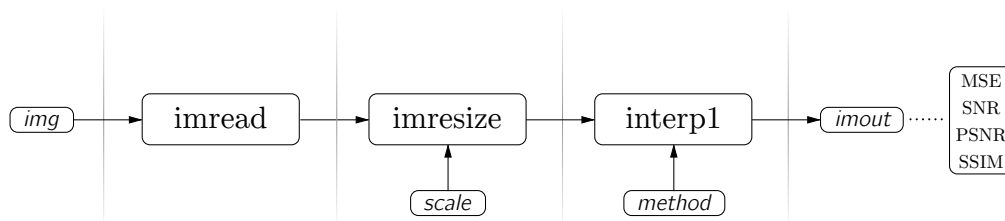
7.1.1 Interpolácia funkciou `interp1`

Pre bežný spôsob interpolácie bola naprogramovaná metóda využitím funkcie `interp1`. Vstup je umožnený pre ľubovoľne veľký obraz ktorý chceme interpolovať. Pre naše účely porovnania pomocou objektívnych metód vstup je vstupný obraz označený na diagrame premenou *img* načítaný pomocou funkcie `imread`, a následne zmenšený premenou definujúcou merítko $scale$ funkciou `imresize`. Takto zmenšený obrázok je následne interpolovaný pomocou funkcie `interp1` najskôr po stĺpcoch a následne po riadkoch. Spôsob interpolácie určuje premenná *method*, ktorá môže nadobúdať argumenty:

- `'nearest'` - interpolácia najbližším susedom (Nespojitá)
- `'linear'` - lineárna interpolácia (C^0)
- `'v5cubic'` - kubická interpolácia konvolúciou (C^1)
- `'pchip'` - polynimálna Hermitovskou kubikou (C^1)
- `'cubic'` - kubická interpolácia konvolúciou, rovnaká ako `pchip` (C^1)

- 'spline' - interpolácia splajnom (C^2)

Výstup sa ukladá do premennej *imout*. Následne sa vypočítajú hodnoty pre objektívne metódy.

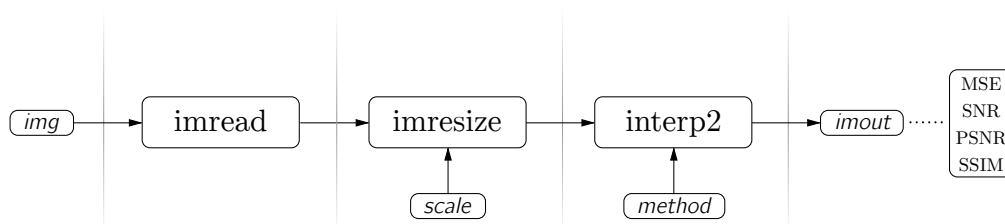


Obr. 7.1: Schéma spracovania obrazu pomocou `interp1`.

7.1.2 Interpolácia funkciou `interp2`

Pre porovnanie bola naprogramovaná i metóda pomocou funkcie `interp2`. Vstup tvoria pôvodné jasové hodnoty 2D obrázku spolu s novými hodnotami určenými pomocou funkcie `meshgrid`. Taktiež spôsob interpolácie určuje premenná *method*, ktorá môže nadobúdať iba argumenty:

- 'nearest' - interpolácia najbližším susedom (Nespojitá)
- 'linear' - lineárna interpolácia (C^0)
- 'cubic' - kubická interpolácia konvolúciou (C^1)
- 'spline' - interpolácia splajnom (C^2)

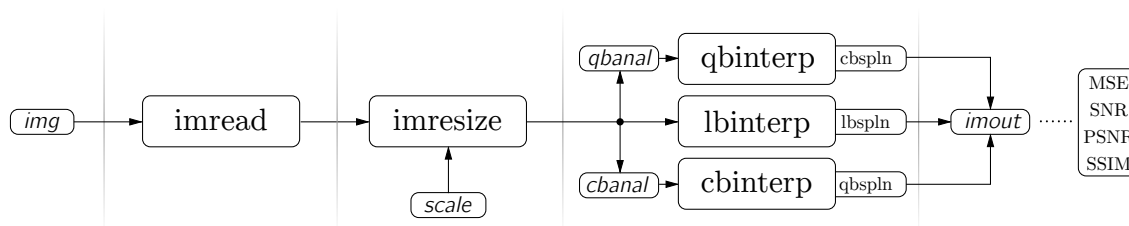


Obr. 7.2: Schéma spracovania obrazu pomocou `interp2`.

Pre skoro totožné výsledky oproti funkcií `interp1` nie je nakoniec zahrnutá v texte porovnávania diplomovej práce.

7.1.3 Interpolácia využitím toolboxu *msplines1*

Pre pokročilú interpoláciu pomocou B-splajnu bol použitý toolbox [37] od *Jána Kybica* z ČVUT. Predspracovanie signálu prebieha obdobne ako v predošlých prípadoch, aby pracoval s rovnakými vstupnými hodnotami ako predchádzajúce metódy. Z toolboxu pre interpoláciu lineárnym B-splajnom sa používa funkcia `lbinterp`, ktorá používa funkciu pre výpočet splajnu `lbspln`, pričom nepotrebuje výpočet bázových funkcií. Naproti tomu interpolácia kvadratickým B-splajnom funkciou `qbinterp`, používajúca pre výpočet splajnu `qbspln` potrebuje výpočet bázových funkcií zo signálu pomocou funkcie `qbanal`. Rovnako pre interpoláciu kubickým B-splajnom funkciou `cbinterp`, používajúcu pre výpočet splajnu `cbspln`, je potrebný výpočet bázových funkcií zo signálu pomocou funkcie `cbanal`. Interpolácia prebieha najskôr po riadkoch, následne po stĺpcoch. Niektoré s funkcií, boli upravené. Takto je možné v konečnom dôsledku interpolovať lineárnym, kvadratickým a kubickým B-splajnom.

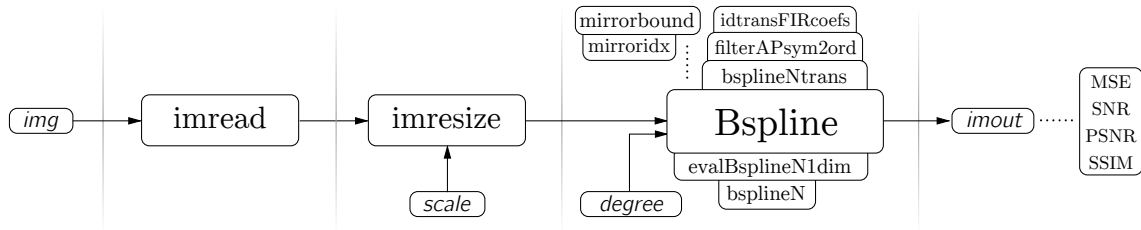


Obr. 7.3: Schéma spracovania obrazu použitím *msplines1*.

7.1.4 Interpolácia využitím toolboxu *B-spline tools 1.2*

Pre ďalší spôsob pokročilejšej interpolácie pomocou B-splajnu bol použitý toolbox [36] od *Jan Tore Korneliussen*. Predspracovanie signálu prebieha obdobne ako v predošlých prípadoch, aby pracoval s rovnakými vstupnými hodnotami ako predchádzajúce metódy. Toolbox bol do značnej miery upravený pre potreby diplomovej práce. Pre výpočet B-splajnu sa používa objektová trieda `Bspline`, využívajúca funkciu `evalBsplineN1dim` pre vyhodnotenie B-splajnu cez 1D dimenziu. Funkcia používa pre výpočet funkciu `bsplineN`, kde sú definované bázové funkcie pre jednotlivé stupne B-splajnu. Upravený toolbox umožňuje výpočet B-splajnu pre stupeň 0 až 4 definovaným premennou *degree*. Funkcia `bsplineNtrans` vypočíta B-spline cez zadané hodnoty používajúc funkciu `filterAPsym2ord`, ktorá predstavuje symetrický filter 2. rádu. Funkcia `idtrasFIRcoefs` je iba tabuľkou s prepočítanými

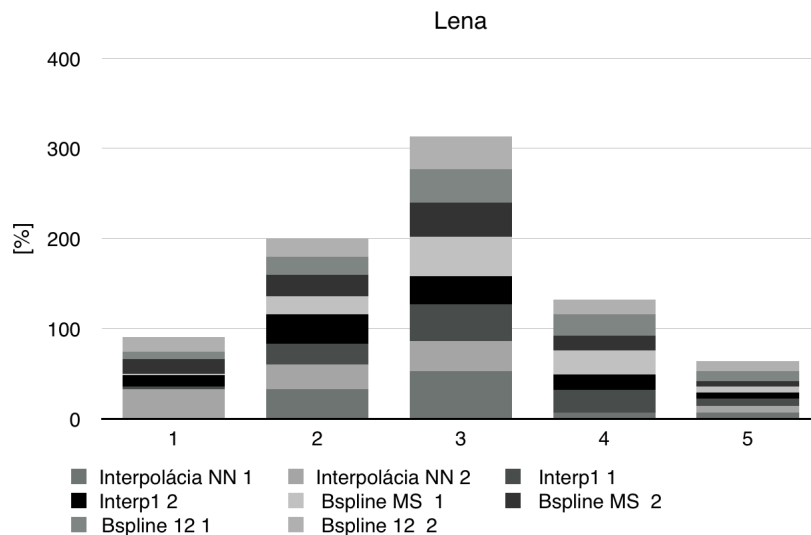
koeficientami pre splajnu nízkeho rádu.



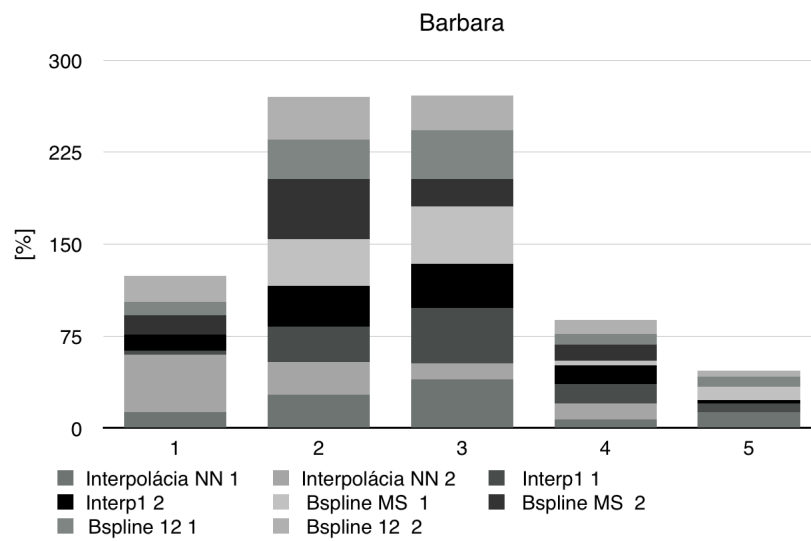
Obr. 7.4: Schéma spracovania obrazu použitím *B-spline tools 1.2*.

7.2 Subjektívne porovnanie metódou DSIS

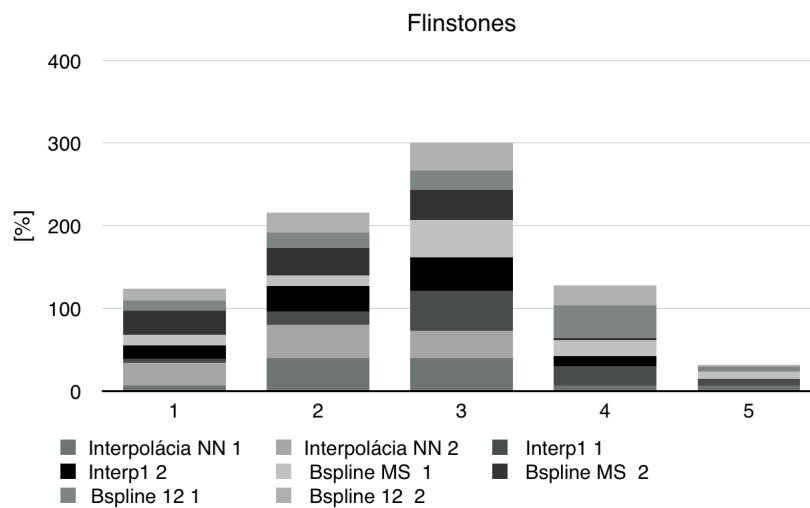
Pre subjektívne porovnanie metódou DSIS boli vybrané obrázky lena.png, barbara.png, flinstones.png a zoneplate.png. Celkový počet hodnotiacich ľudí bol minimálne 15 podľa doporučenie ITU-R BT.500-11. Rozsah možného hodnotenia bol 1 až 5, kde 1 značí najmenšiu zhodu a 5 zhodu najväčšiu. Grafy sú stĺpcové vyjadrujúce percentuálne zhodnotenie výsledkov. Index 1 je pre stanovenú otázku: „Nakoľko je sa zhoduje spracovaný obrázok s referenčným obrázkom?“ a index 2 pre stanovenú otázku „Ako subjektívne pôsobí spracovaný obrázok?“.



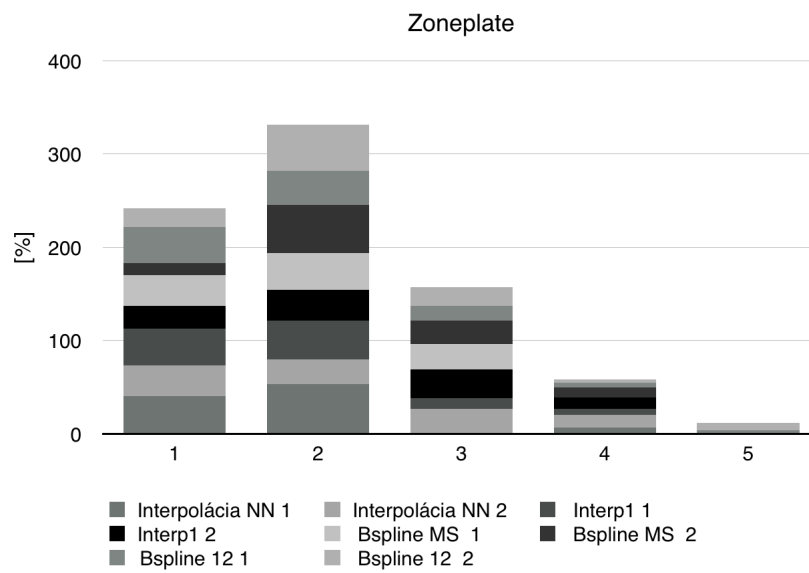
Obr. 7.5: Graf výsledkou metódy DSIS pre obrázok lena.png.



Obr. 7.6: Graf výsledkou metódy DSIS pre obrázok barbara.png.



Obr. 7.7: Graf výsledkou metódy DSIS pre obrázok flinstones.png.



Obr. 7.8: Graf výsledkou metódy DSIS pre obrázok zoneplate.png.

8 ZÁVER

Cielom diplomovej práce bolo teoreticky popísať a naprogramovať rôzne spôsoby interpolácie obrazových signálov bežnými a pokročilými metódami. Potom následne jednotlivé metódy pomocou objektívnych a subjektívnych metód hodnotenia porovnať.

Prvú časť diplomovej práce tvorí bežný úvod do teórie počítačovej grafiky. Rozobrané sú druhy počítačovej grafiky, ľudské videnie a farebné modely. Druhá časť sa zaoberá problematikou súvisiacou s digitálnym spracovaním obrazu. Vysvetľuje digitalizáciu obrazových signálov, ich reprezentáciu a spracovanie v počítačových systémoch. Tretiu časť tvorí úvod do teórie analytických kriviek v rovine. Uvádza spôsoby ich vyjadrenia a zápisu spolu s definíciou požadovaných vlastností.

Ďalšia kapitola sa už konečne venuje interpolácií obrazu, kde na teoretickom základe zo získaných informácií a dostupných materiálov zhŕňa bežné spôsoby interpolácie obrazu použitím tzv. konvolučného jadra. Rekonštrukcia po častiach spojitých funkcie sa získavala z diskretných hodnôt ako lineárna kombinácia vstupného obrazu a rekonštrukčného filtra. Pre jednoduchosť pochopenia a vizualizácie boli všetky konvolučné jadra separované a vysvetlené najskôr pre 1D a až potom zovšeobecnené do 2D pre spracovanie obrazu. Ďalšia kapitola pristupuje najskôr k definícií predchodcov B-splajn kriviek, ktorými sú Fergusonove, Bézierove či Coonsove krivky. A až potom výsledne k B-splajnom a ich dvojitému zovšeobecneniu NURBS, definovaných bazovými B-splajn funkciami. V závere je rozobraný problém interpolácie pomocou NURBS kriviek a jeho riešenie.

V programovom prostredí MATLAB[®] boli naprogramované riešenia ako pre bežné spôsoby interpolácie tak i pokročilé. Taktiež bolo naprogramovaných niekoľko rôznych demonštrácií súvisiacich s diplomovou prácou. Naprogramované boli jednotlivé konvolučné jadrá pre bežné metódy interpolácie či interpolácia najbližším susedom. Ďalej boli naprogramované demonštrácie pre bazové funkcie Bernsteinových, Hermitových a Langragerových polynómov. Taktiež demonštrácia bazových funkcií pre B-splajny. Prípadne demonštrácie pre jednoduché Bézierove krivky a taktiež i pre racionálne Bézierove krivky s pridaním váh. Poprípade demonštrácie použité v rámci diplomovej práce.

Pre bežné metódy interpolácie boli využité funkcie `interp1` a `interp2` s definovanými spôsobmi interpolácie. Bližší popis jednotlivých častí a spôsob interpolácie je popísaný v poslednej časti diplomovej práce. V spracovaných výsledkoch textu diplomovej práce bola zahrnutá iba `interp1` z dôvodu priam totožných výsledkov pre bežné metódy. Pre pokročilé metódy interpolácie boli využité dva externé toolboxy pre modelovanie obecných B-splajn kriviek. Funkcie toolboxov boli upravené práve pre optimalizovanie interpolácie obrazového signálu. Prvým toolboxom je `msplines1`

[37] od *Jána Kybica*, ktorým je možné generovať B-splajnamy druhého až tretieho stupňa. V prípade upraveného druhého toolboxu *msplines1* od *Jan Tore Korneliusen* je možné generovať B-splajn krivky od nultého do štvrtého stupňa.

Výsledné funkčné metódy interpolácie obrazu boli aplikované na 20 rôznych obrázkov, ktorých iba časť je v prílohe diplomovej práce. Všetky obrázky boli zhodnotené v programe MATLAB[®] pomocou popísaných objektívnych metód. Taktiež bola spracovaná subjektívna metóda DSIS podľa doporučenie ITU-R BT.500-11.

V diplomovej práci sa však nakoniec nepodarilo úspešne a funkčne naprogramovať interpolačnú metódu krivkami NURBS. Všetky programové riešenia viac či menej, boli iba aproximáciami čím sa vo veľkej miere strácala informácia z pôvodného „interpolovaného obrazu“. Aproximačné krivky podávali zakaždým v konečnom dôsledku značne špatné výsledky testovaného obrazu, a preto z toho dôvodu ani neboli zahrnuté vo výslednej diplomovej práci. Samotné toolboxy vychádzajúce z knihy *The NURBS Book* [23] neoplývali funkciou pre výpočet sústavy rovníc riadiacich bodov aby daná krivka prechádzala bodmi definičnými. Následný pokus vlastnej implementácie algoritmu a jeho použitia z knihy *The NURBS Book* skončil neúspechom. Problém totiž zjavne spočíval ako sa domnievam skôr v nekonzistencii implementovaného algoritmu z knihy s použitými toolboxami pre NURBS, ako z vlastnej nesprávnej funkčnosti.

LITERATÚRA

- [1] AKIMA, H. *A Method of Smooth Curve Fitting* ESSA Tech. Rep. ERL 101-ITS 73 (Washington, D.C.: U.S. Government Printing Office), 1969.
- [2] *Barvy* [online]. Dostupné z URL: <<http://www.geografie.webzdarma.cz/-barvy.pdf>> [cit. 8.11.2013].
- [3] *Binocular vision* [online]. Dostupné z URL: <http://en.wikipedia.org/wiki/Binocular_vision> [cit. 11.4.2014].
- [4] *Color models* [online]. Dostupné z URL: <http://en.wikipedia.org/wiki/Color_models> [cit. 8.11.2013].
- [5] ČIHÁH, M. *Lagrangeovy polynomy* [online]. Dostupné z URL: <http://lide.uhk.cz/prf/ucitel/cihakmi1/zanumat/lagrangeovy_polynomy.pdf> [cit. 5.3.2014].
- [6] CHENG, Y. *Edge-Adaptive Image Interpolation Techniques* [online]. Department of Electrical Engineering, Stanford University. Dostupné z URL: <<http://scien.stanford.edu/pages/labsite/2001/ee368/projects2001/dropbox/-project04/>> [cit. 15.11.2013].
- [7] EBERLY, D. *Kochanek-Bartels Cubic Splines (TCB Splines)* [online]. Geometric Tools, LLC, February 14, 2008. Dostupné z URL: <<http://www.geometrictools.com/Documentation/KBSplines.pdf>> [cit. 4.3.2014].
- [8] DOBEŠ, M. *Zpracování obrazu a algoritmy v C#*. BEN - Technická literatura, 1. vyd., Praha 2008, ISBN 987-80-7300-233-6.
- [9] DODGSON, A. N. *Image Resampling* [online]. Technická správa č. 261, Wolfson College & Computer Laboratory, University Cambridge, Cambridge, U.K., Aug, 1992, ISSN 1476-2986. Dostupné z URL: <<http://www.cl.cam.ac.uk/techreports/UCAM-CL-TR-261.pdf>> [cit. 13.11.2013].
- [10] DODGSON, A. N. Quadratic Interpolation for Image Resampling. *IEEE Transactions on Image Processing* [online]. 1997, vol. 6, no. 9. Dostupné z URL: <<http://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=00623195&tag=1>> [cit. 5.11.2013].
- [11] FERGUSON, C. J. *Multivariable Curve Interpolation* In Journal of the ACM (JACM), Volume 11 Issue 2, April 1964.

- [12] FRASER, B.; MURPHY, CH.; BUNTING, F. *Správa barev* Computer Press, Brno, 2003. ISBN 80-722- 6943-7.
- [13] GONZALES, C. R.; WOODS, E.R. *Digital Image Processing [3rd Edition]*. Upper Saddle River: Prentice Hall, 3rd ed., august 2007, 976 s. ISBN 978-0131687288 .
- [14] GOURGOULHON, E. *An introduction to polynomial interpolation* [online]. 14-18 November 2005, Meudon Observatory, France Dostupné z URL: <<http://www.lorene.obspm.fr/school/polynom.pdf>> [cit. 5.3.2014].
- [15] JOY, I. K. *Catmull-Rom Splines* [online]. Visualization and Graphics Research Group Department of Computer Science University of California, Davis, 2002. Dostupné z URL: <<http://graphics.cs.ucdavis.edu/~joy/ecs278/notes/Catmull-Rom-Spline.pdf>> [cit. 5.3.2014].
- [16] LINKEOVÁ, I. *NURBS křivky (NeUniformní Racionální B-Spline křivky)*. 1. vyd. Praha: Nakladatelství ČVUT, 2007, 208 s., ISBN 978-80-01-03893-2.
- [17] LEHMANN, T. M.; GONNER, C.; SPITZER, K. Survey: Interpolation Methods in Medical ImageProcessing. *IEEE Transactions on Medical Imaging* [online]. 1999, vol. 18, p. 1049 - 1075. Dostupné z URL: <<http://kwon.dongseo.ac.kr/~lbg/cagd/kmmcs/200608/lehmann.pdf>>. [cit. 28.11.2013].
- [18] MORAVEC, Z. *Fraktály* [online]. 2006, Bakalářská práce. Masarykova univerzita, Přírodovědecká fakulta. Vedoucí práce Ladislav Adamec. Dostupné z URL: <http://is.muni.cz/th/106681/prif_b/> [cit. 15.12.2013].
- [19] PAN, J. *Image Interpolation using Spline Curves* [online]. Department of Mechanical Engineering, 6. Decembra, 2003. Dostupné z URL: <http://cadcam.eng.sunysb.edu/~purwar/Teaching/MEC572/Term_Papers/-Jiahui_Pan_TermPaper.pdf> [cit. 20.12.2013].
- [20] PARKER, A. J.; KENYON, R.V.; TROXEL, E.D. Comparison of interpolation methods for image resampling. *IEEE Transaction on Medical Imaging* [online]. Vol. Mi-2, No. 1, Mar, 1983, pp. 31-39. Dostupné z URL: <<http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=4307610>>.
- [21] PASCAL, G. *Linear Methods for Image Interpolation* [online]. Posledná aktualizácia 27.9.2011 [cit. 12.11.2013], IPOL Image Processing On Line, ISSN 2105-1232. Dostupné z URL: <http://www.ipol.im/pub/art/2011/g_lmii/>.
- [22] PIEGL, L. Modifying the shape of rational B-splines Part 1: curves. *Computer Aided Design*.. 1989, 21, s. 509-18.

- [23] PIEGL, L.; TILLER, W. *The NURBS Book*. Monographs in Visual Communications Springer, Berlin, 2nd edition, 1997, ISBN 3-540-61545-8.
- [24] PRAUTZSCH, H.; BOEH, W.; PALUSZNY, M. *Bézier and B-Spline Techniques*. Springer, 2002.
- [25] PROCHÁZKOVÁ, J. *Modelování matematických ploch v CAD systémech*. Disertačná práca. 2007. Dostupné z URL: <https://akela.mendelu.cz/xproch17/jana/prochazkova_disertace_text.pdf>.
- [26] RAJMIC, P.; SCHIMMEL, J. *Moderní počítačová grafika*. Ústav telekomunikací, VUT Brno FEKT, 1. vydanie, 2013.
- [27] *Recommendation ITU-R BT.500-11 - Methodology for the subjective assessment of the quality of television pictures* [online]. June, 2006. Dostupné z URL: <http://www.itu.int/dms_pubrec/itu-r/rec/bt/R-REC-BT.500-11--200206-S!!PDF-E.pdf>.
- [28] SOJKA, E., GAURA, J., KRUMNIKL, M. *Matematické základy digitálního zpracování obrazu* [online]. VŠB Ostrava, ZČU v Plzni, 2011. Dostupné z URL: <http://mi21.vsb.cz/sites/mi21.vsb.cz/files/unit/digitalni_zpracovani_obrazu.pdf> [cit. 12.12.2013].
- [29] THÉVENAZ P.; BLU T.; UNSER M. *Image Interpolation and Resampling*. Dostupné z URL: <<http://bigwww.epfl.ch/publications/thevenaz9901.pdf>>.
- [30] WANG, Z.; BOVIK, A.C. Mean Squared Error: Love it or leave it?. *IEEE Signal Processing Magazine* [online]. January 2009. Dostupné z URL: <http://live.ece.utexas.edu/publications/2009/zw_spm_jan09.pdf> [cit. 7.12.2013].
- [31] WANG Z.; BOVIK A.C.; SHEIKH H.R.; SIMONCELLI E.P. Image quality assessment: From error visibility to structural similarity. *IEEE Transactions on Image Processing* [online]. Vol. 13, no.4, april 2004. Dostupné z URL: <<https://ece.uwaterloo.ca/~z70wang/publications/ssim.pdf>> [cit. 7.12.2013].
- [32] WANG Z.; BOVIK A.C.; SHEIKH H.R.; SIMONCELLI E.P. *The SSIM Index for Image Quality Assessment* [online]. Posledná aktualizácia 20.1.2012 [cit. 7.12.2013]. Dostupné z URL: <<https://ece.uwaterloo.ca/~z70wang/research/ssim/>>.
- [33] WOLBERG, G. Digital Image Warping. *IEEE Computer Society Press*. Los Alamitos, CA, 1990, s.318.

- [34] ŽÁRA, J.; BENEŠ, B.; SOCHAR, J.; FELKEL, P. *Moderní počítačová grafika*. Computer Press, druhé vydanie, 2005, ISBN 80-251-0454-0.
- [35] ZHANG, X.; WU, X. *Image Interpolation by Adaptive 2-D Autoregressive Modeling and Soft-Decision Estimation* [online]. Dostupné z URL: <<http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.141.1589&rep=rep1&type=pdf>> [cit. 16.11.2013].

KNÍŽNICE PROGRAMU MATLAB[®]

- [36] KORNELIUSSEN, T. J. *B-spline tools* [knížnica programu MATLAB[®]]. Ver. 1.2, May 2011. Dostupné z URL: <<http://www.mathworks.ch/matlabcentral/fileexchange/27047-b-spline-tools/content/bspline4.m>> [cit. 20.5.2014].
- [37] KYBIC, J. *Matlab B-spline Routines Repository* [knížnica programu MATLAB[®]]. Ver. 1.0, April 2008. Dostupné z URL: <<http://cmp.felk.cvut.cz/~kybic/old/msplines/list.html>> [cit. 21.5.2014].
- [38] FALCO, C.; VAZQUEZ, R. *NURBS* [knížnica programu MATLAB[®]]. Ver. 1.3.7, 28. April 2013. Dostupné z URL: <<http://octave.sourceforge.net/nurbs/>> [cit. 21.5.2014].
- [39] ZHANG, P. *NURBS Toolbox by D.M. Spink* [knížnica programu MATLAB[®]]. Ver. 1.0, 15. Jan 2010. Dostupné z URL: <<http://www.mathworks.com/matlabcentral/fileexchange/26390-nurbs-toolbox-by-d-m-spink>> [cit. 23.5.2014].
- [40] WANG, Z. *Structural SIMilarity index* [funkcia programu MATLAB[®]]. Ver. 1.0, 2003. Dostupné z URL: <<https://ece.uwaterloo.ca/~z70wang/research/ssim/ssim.m>> [cit. 16.5.2014].

ZOZNAM SYMBOLOV, VELIČÍN A SKRATIEK

- * konvolúcia
- 1D (*One Dimensional*) jednorozmerný
- 2D (*Two Dimensional*) dvojrozmerný
- 3D (*Three Dimensional*) trojrozmerný
- a parameter
- b veľkosť v bitoch
- B (*Brightness*) svetlosť
- B_i^n bernsteinovy polynómy n -tého stupňa
- C** matica parametrov krivky
- C_i coonsovy polynómy
- C^n parametrická spojitosť
- CMY (*Cyan, Magenta, Yellow*) azúrová, fialová, žltá
- EM elektromagnetické
- f_{max} najvyššia frekvencia
- F_i hermitovské polynómy
- F_n diskrétna fourierova 1D transformácia
- f_s vzorkovacia frekvencia
- $F(u)$ spojitá fourierova 1D transformácia
- $F(u, v)$ spojitá fourierova 2D transformácia
- $f(x)$ spojitá 1D funkcia
- $f(x, y)$ spojitá 2D funkcia
- G** vektor geometrických podmienok
- G^n geometrická spojitosť
- $h_{i,j}$ konvolučná maska

$h(x)$ konvolučné 1D jadro
 $^{\text{Ideal}}h(x)$ konvolučné jadro pre ideálnu interpoláciu
 $^{\text{Sinc}}h_N(x)$ konvolučné jadro pre sinc interpoláciu
 $^{\text{Sused}}h_1(x)$ konvolučné jadro pre interpoláciu najbližším susedom
 $^{\text{Linear}}h_2(x)$ konvolučné jadro pre lineárnu interpoláciu
 $^{\text{Kvadr}}h_3(x)$ konvolučné jadro pre kvadratickú interpoláciu
 $^{\text{Kubic}}h_4(x)$ konvolučné jadro pre kubickú interpoláciu
 $h(x, y)$ konvolučné 2D jadro
HLS (*Hue, Lightnes, Saturation*) odtien, svetlosť, sýtosť
HSB (*Hue, Saturation, Brightness*) odtieň, sýtosť, jas
HVS (*Human Visual System*) ľudský systém videnia
I (*Intensity*) jas
 I_i diskretná 1D funkcia
 I_k diskretná 1D funkcia
 $I_{i,j}$ diskretná 2D funkcia
 L stupne šedi
 \mathbf{M} bazová matica
 M, N rozmery hodnôt obrazu
MSE (*Mean Square Error*) stredná kvadratická chyba
 n stupeň
 \mathbb{N} množina prirodzených čísiel
 \mathbb{N}_0 množina prirodzených čísiel s nulou
 N_i^k b-splajn bazová funkcia stupňa k
NN (*Nearest neighbor*) najbližší sused
NURBS (*Non-uniformal Rational Basis Spline*) neuniformný racionálny b-splajn

\mathbf{P}_i riadiace body krivky
 \mathbf{p}'_i dotyčnicový vektor v riadiacom bode
 PSNR (*Peak Signal to Noise Ratio*) špičkový pomer signálu k šumu
 \mathbb{R} množina reálnych čísiel
 RGB (*Red, Green, Blue*) červená, zelená, modrá
 S (*Saturation*) sýtosť
 SSIM (*Structural Similarity*) štrukturálna podobnosť
 t čas
 \mathbf{t} uzlový vektor
 \mathbf{T} matica času
 $Q(t)$ všeobecná krivka
 $Q_n(t)$ polynóm stupňa n
 ${}^{\text{B-splajn}}Q(t)$ b-splajn krivka
 ${}^{\text{Coons}}Q(t)$ coonsova krivka
 ${}^{\text{Bézier}}Q(t)$ bézierová krivka
 ${}^{\text{Hermit}}Q(t)$ hermitovská krivka
 ${}^{\text{NURBS}}Q(t)$ NURBS krivka
 \mathbf{q}' dotyčnicový vektor
 \mathbf{q}'_i dotyčnicový vektor v uzle
 $w(x)$ okienková funkcia
 \mathbb{Z} množina celých čísiel
 ξ, η súradnice

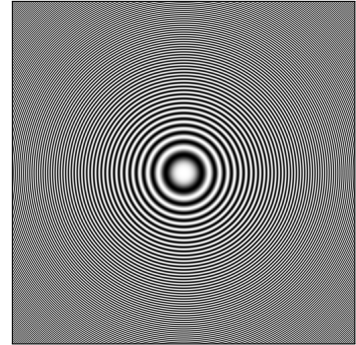
A VÝSLEDNÉ TESTOVANÉ OBRAZY MATLAB



(a) lena.png



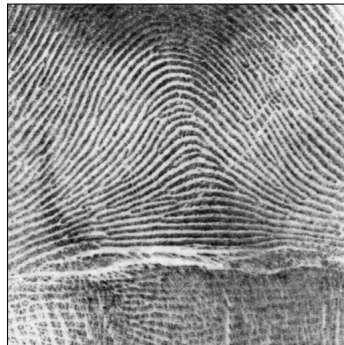
(b) barbara.png



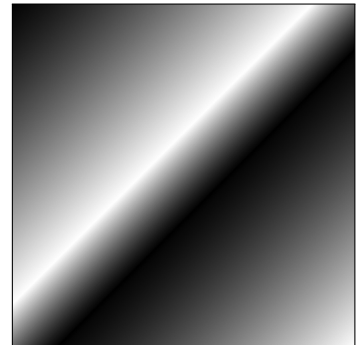
(c) zoneplane.png



(d) flinstones.png



(e) fingerprint.png



(f) gradient.png



(g) cameraman.png

Obr. A.1: Časť testovacích obrázkov pre demonštráciu.

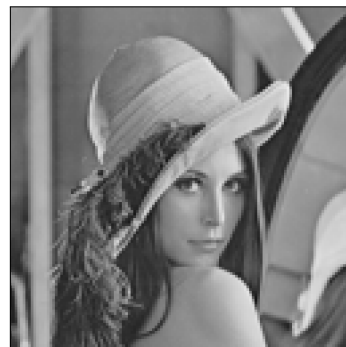
Obr. A.2: Obrázok lena.png.



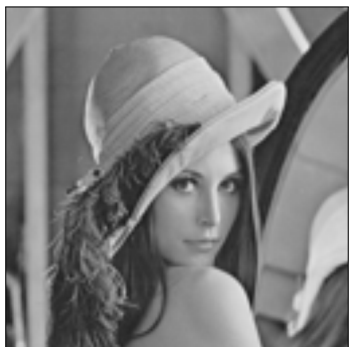
(a) Originálny obrázok



(b) Metoda NN
MSE = 0.0020572
SNR = 12.301 dB
PSNR = 26.8672 dB
SSIM = 0.88633



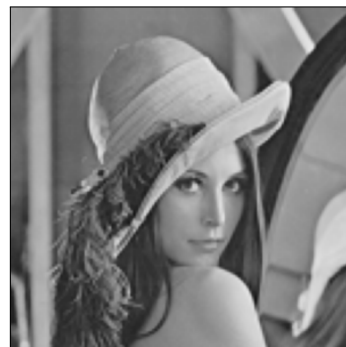
(c) Metoda nearest
MSE = 0.002543
SNR = 11.3804 dB
PSNR = 25.9466 dB
SSIM = 0.87824



(d) Metoda linear
MSE = 0.001955
SNR = 12.5223 dB
PSNR = 27.0885 dB
SSIM = 0.87603



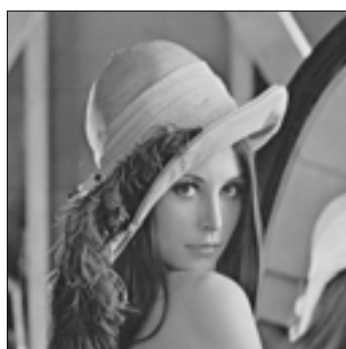
(e) Metoda v5cubic
MSE = 0.0017673
SNR = 12.9607 dB
PSNR = 27.5269 dB
SSIM = 0.89207



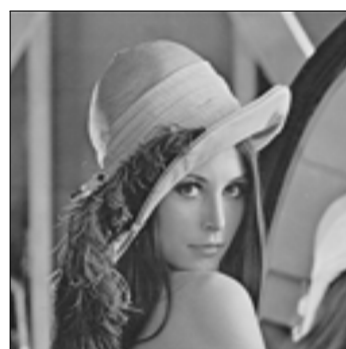
(f) Metoda pchip
MSE = 0.0017643
SNR = 12.968 dB
PSNR = 27.5342 dB
SSIM = 0.89093



(g) Metoda spline
MSE = 0.001709
SNR = 13.1065 dB
PSNR = 27.6727 dB
SSIM = 0.89702



(h) Metoda lineaBspline
MSE = 0.001955
SNR = 12.5223 dB
PSNR = 27.0885 dB
SSIM = 0.87603



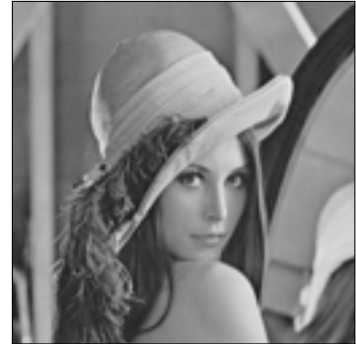
(i) Metoda kvadrBspline
MSE = 0.0018764
SNR = 12.7005 dB
PSNR = 27.2667 dB
SSIM = 0.87603



(j) Metoda cubicBspline
MSE = 0.0018738
SNR = 12.7065 dB
PSNR = 27.2727 dB
SSIM = 0.87603



(k) Metoda Bspline0
MSE = 0.002543
SNR = 11.3804 dB
PSNR = 25.9466 dB
SSIM = 0.87824



(l) Metoda Bspline1
MSE = 0.001955
SNR = 12.5223 dB
PSNR = 27.0885 dB
SSIM = 0.87603



(m) Metoda Bspline2
MSE = 0.00172
SNR = 13.0786 dB
PSNR = 27.6448 dB
SSIM = 0.89614

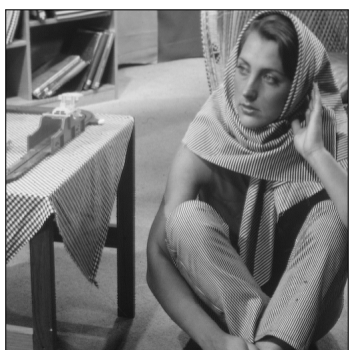


(n) Metoda Bspline3
MSE = 0.0017069
SNR = 13.1116 dB
PSNR = 27.6778 dB
SSIM = 0.89702



(o) Metoda Bspline4
MSE = 0.0016902
SNR = 13.1543 dB
PSNR = 27.7205 dB
SSIM = 0.89847

Obr. A.3: Obrázok barbara.png.



(a) Originálny obrázok



(b) Metoda NN
MSE = 0.0051543
SNR = 9.4585 dB
PSNR = 22.8783 dB
SSIM = 0.78295



(c) Metoda nearest
MSE = 0.0058126
SNR = 8.9366 dB
PSNR = 22.3563 dB
SSIM = 0.77145



(d) Metoda linear
MSE = 0.0052219
SNR = 9.402 dB
PSNR = 22.8217 dB
SSIM = 0.771



(e) Metoda v5cubic
MSE = 0.0050828
SNR = 9.5192 dB
PSNR = 22.9389 dB
SSIM = 0.78543



(f) Metoda pchip
MSE = 0.0050918
SNR = 9.5115 dB
PSNR = 22.9313 dB
SSIM = 0.78475



(g) Metoda spline
MSE = 0.0050491
SNR = 9.5481 dB
PSNR = 22.9678 dB
SSIM = 0.79029



(h) Metoda lineaBspline
MSE = 0.0052219
SNR = 9.402 dB
PSNR = 22.8217 dB
SSIM = 0.771



(i) Metoda kvadrBspline
MSE = 0.0051385
SNR = 9.4719 dB
PSNR = 22.8916 dB
SSIM = 0.771



(j) Metoda cubicBspline
MSE = 0.005133
SNR = 9.4766 dB
PSNR = 22.8963 dB
SSIM = 0.771



(k) Metoda Bspline0
MSE = 0.005812
SNR = 8.9366 dB
PSNR = 22.3563 dB
SSIM = 0.77145



(l) Metoda Bspline1
MSE = 0.0052219
SNR = 9.402 dB
PSNR = 22.8217 dB
SSIM = 0.771



(m) Metoda Bspline2
MSE = 0.0050216
SNR = 9.5718 dB
PSNR = 22.9915 dB
SSIM = 0.78913

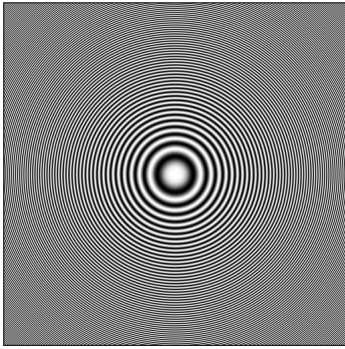


(n) Metoda Bspline3
MSE = 0.005004
SNR = 9.5871 dB
PSNR = 23.0068 dB
SSIM = 0.79032

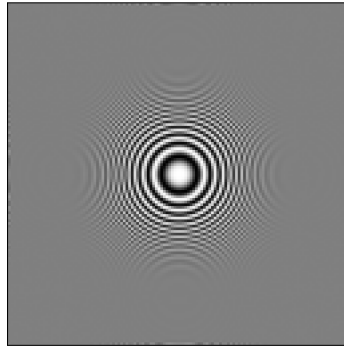


(o) Metoda Bspline4
MSE = 0.004989
SNR = 9.6002 dB
PSNR = 23.0199 dB
SSIM = 0.79166

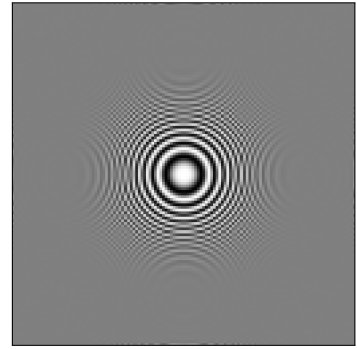
Obr. A.4: Obrázok zoneplate.png.



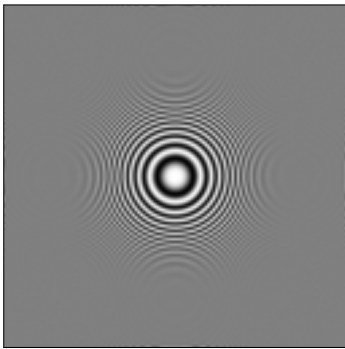
(a) Originálny obrázok



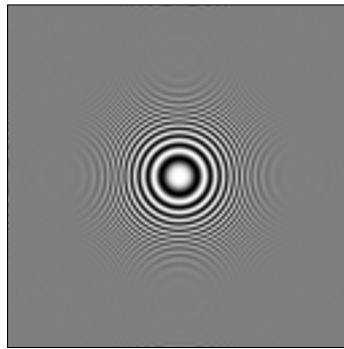
(b) Metoda NN
MSE = 0.11223
SNR = 0.43506 dB
PSNR = 9.4988 dB
SSIM = 0.15593



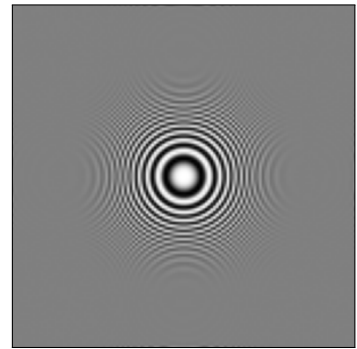
(c) Metoda nearest
MSE = 0.11437
SNR = 0.35292 dB
PSNR = 9.4167 dB
SSIM = 0.12904



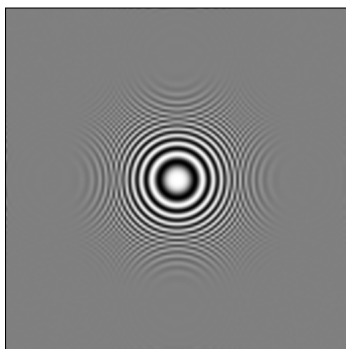
(d) Metoda linear
MSE = 0.11356
SNR = 0.38381 dB
PSNR = 9.4476 dB
SSIM = 0.11814



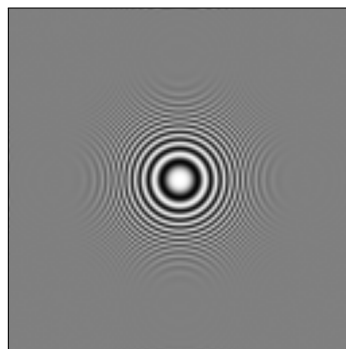
(e) Metoda v5cubic
MSE = 0.11227
SNR = 0.43351 dB
PSNR = 9.4973 dB
SSIM = 0.1316



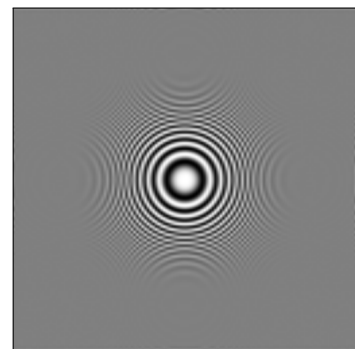
(f) Metoda pchip
MSE = 0.11253
SNR = 0.42337 dB
PSNR = 9.4872 dB
SSIM = 0.12956



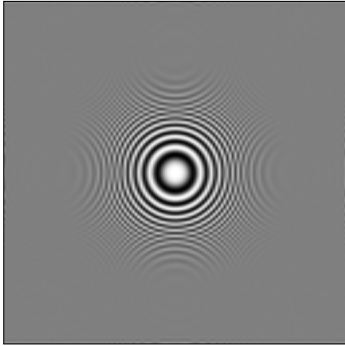
(g) Metoda spline
MSE = 0.11189
SNR = 0.44838 dB
PSNR = 9.5122 dB
SSIM = 0.13503



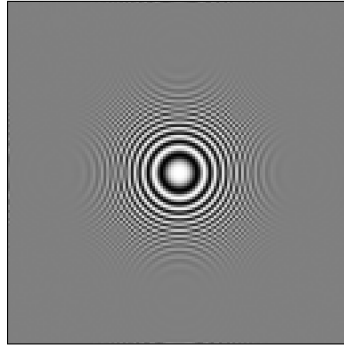
(h) Metoda lineaBspline
MSE = 0.11356
SNR = 0.38381 dB
PSNR = 9.4476 dB
SSIM = 0.11814



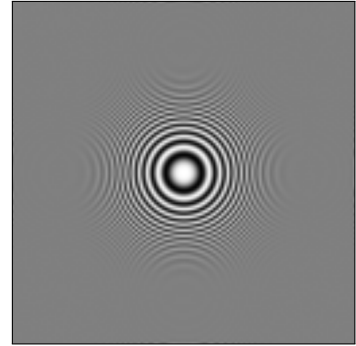
(i) Metoda kvadrBspline
MSE = 0.11274
SNR = 0.41525 dB
PSNR = 9.479 dB
SSIM = 0.11814



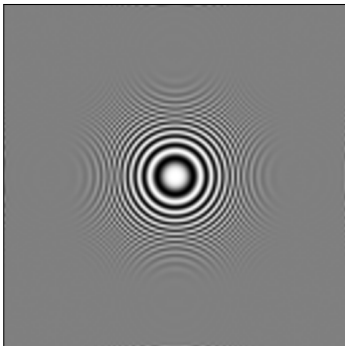
(j) Metoda cubicBspline
MSE = 0.11273
SNR = 0.41591 dB
PSNR = 9.4797 dB
SSIM = 0.11814



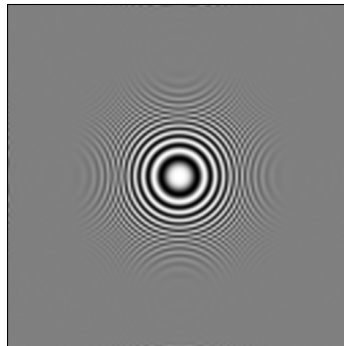
(k) Metoda Bspline0
MSE = 0.11437
SNR = 0.35292 dB
PSNR = 9.4167 dB
SSIM = 0.12904



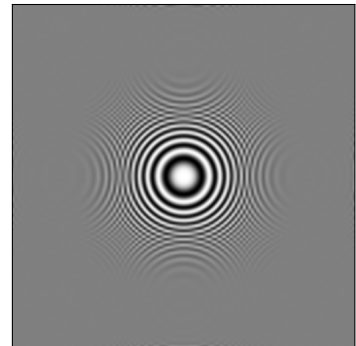
(l) Metoda Bspline1
MSE = 0.11356
SNR = 0.38381 dB
PSNR = 9.4476 dB
SSIM = 0.11814



(m) Metoda Bspline2
MSE = 0.11192
SNR = 0.447 dB
PSNR = 9.5108 dB
SSIM = 0.13551



(n) Metoda Bspline3
MSE = 0.1119
SNR = 0.44788 dB
PSNR = 9.5117 dB
SSIM = 0.13499



(o) Metoda Bspline4
MSE = 0.11178
SNR = 0.4525 dB
PSNR = 9.5163 dB
SSIM = 0.13648

Obr. A.5: Obrázok flinstones.png.



(a) Originálny obrázok



(b) Metoda NN
MSE = 0.012219
SNR = 8.7063 dB
PSNR = 19.1295 dB
SSIM = 0.78754



(c) Metoda nearest
MSE = 0.014966
SNR = 7.8257 dB
PSNR = 18.2488 dB
SSIM = 0.75481



(d) Metoda linear
MSE = 0.012872
SNR = 8.4805 dB
PSNR = 18.9037 dB
SSIM = 0.73548



(e) Metoda v5cubic
MSE = 0.011881
SNR = 8.8285 dB
PSNR = 19.2516 dB
SSIM = 0.77075



(f) Metoda pchip
MSE = 0.012001
SNR = 8.7846 dB
PSNR = 19.2077 dB
SSIM = 0.76811



(g) Metoda spline
MSE = 0.011592
SNR = 8.9352 dB
PSNR = 19.3584 dB
SSIM = 0.78155



(h) Metoda lineaBspline
MSE = 0.012872
SNR = 8.4805 dB
PSNR = 18.9037 dB
SSIM = 0.73548



(i) Metoda kvadrBspline
MSE = 0.012192
SNR = 8.7161 dB
PSNR = 19.1393 dB
SSIM = 0.73548



(j) Metoda kubicBspline
MSE = 0.012153
SNR = 8.7299 dB
PSNR = 19.1531 dB
SSIM = 0.73548



(k) Metoda Bspline0
MSE = 0.014966
SNR = 7.8257 dB
PSNR = 18.2488 dB
SSIM = 0.75481



(l) Metoda Bspline1
MSE = 0.012872
SNR = 8.4805 dB
PSNR = 18.9037 dB
SSIM = 0.73548



(m) Metoda Bspline2
MSE = 0.011524
SNR = 8.9608 dB
PSNR = 19.384 dB
SSIM = 0.77954

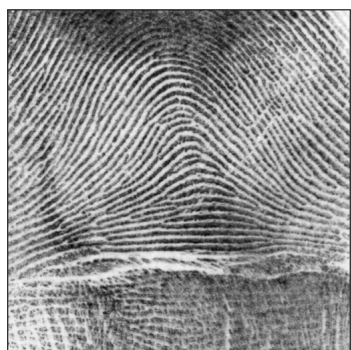


(n) Metoda Bspline3
MSE = 0.011431
SNR = 8.9959 dB
PSNR = 19.419 dB
SSIM = 0.7817

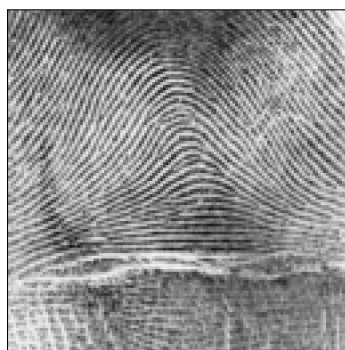


(o) Metoda Bspline4
MSE = 0.011328
SNR = 9.0352 dB
PSNR = 19.4584 dB
SSIM = 0.78459

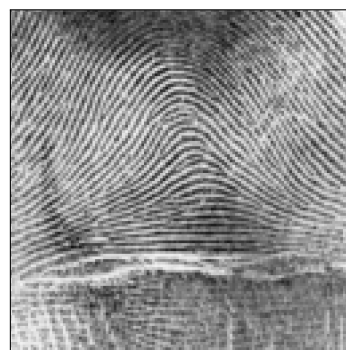
Obr. A.6: Obrázok fingerprint.png.



(a) Originálny obrázok



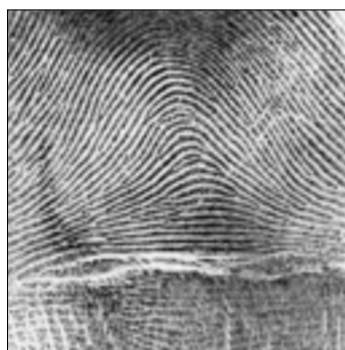
(b) Metoda NN
MSE = 0.010096
SNR = 5.9774 dB
PSNR = 19.9587 dB
SSIM = 0.78288



(c) Metoda nearest
MSE = 0.013834
SNR = 4.609 dB
PSNR = 18.5904 dB
SSIM = 0.70466



(d) Metoda linear
MSE = 0.011007
SNR = 5.6018 dB
PSNR = 19.5831 dB
SSIM = 0.66881



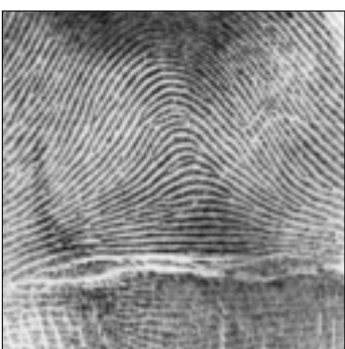
(e) Metoda v5cubic
MSE = 0.0094952
SNR = 6.2436 dB
PSNR = 20.225 dB
SSIM = 0.74155



(f) Metoda pchip
MSE = 0.010062
SNR = 5.9917 dB
PSNR = 19.9731 dB
SSIM = 0.72018



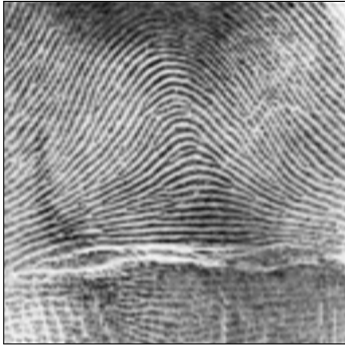
(g) Metoda spline
MSE = 0.0089281
SNR = 6.511 dB
PSNR = 20.4924 dB
SSIM = 0.76911



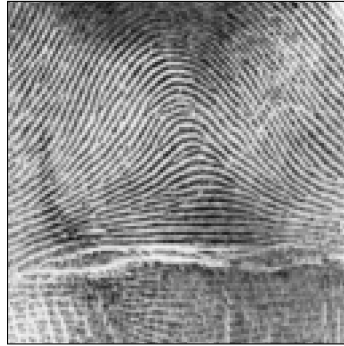
(h) Metoda lineaBspline
MSE = 0.011007
SNR = 5.6018 dB
PSNR = 19.5831 dB
SSIM = 0.66881



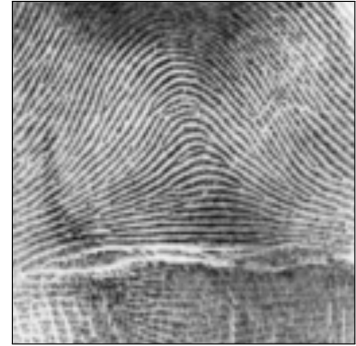
(i) Metoda kvadrBspline
MSE = 0.0096834
SNR = 6.1584 dB
PSNR = 20.1397 dB
SSIM = 0.66881



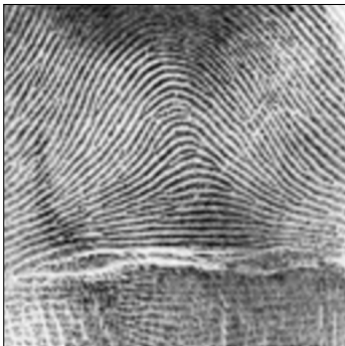
(j) Metoda kubicBspline
MSE = 0.0095418
SNR = 6.2223 dB
PSNR = 20.2037 dB
SSIM = 0.66881



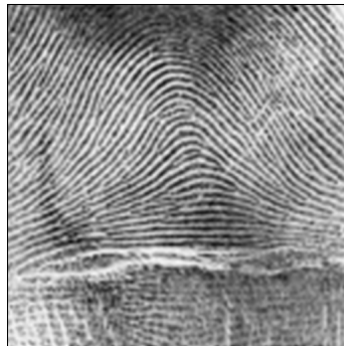
(k) Metoda Bspline0
MSE = 0.013834
SNR = 4.609 dB
PSNR = 18.5904 dB
SSIM = 0.70466



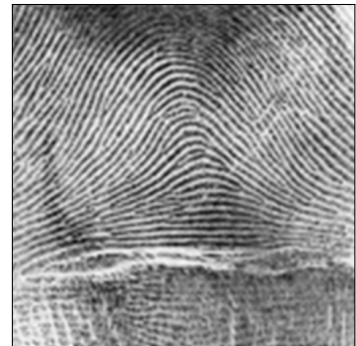
(l) Metoda Bspline1
MSE = 0.011007
SNR = 5.6018 dB
PSNR = 19.5831 dB
SSIM = 0.66881



(m) Metoda Bspline2
MSE = 0.0090015
SNR = 6.4755 dB
PSNR = 20.4569 dB
SSIM = 0.7616

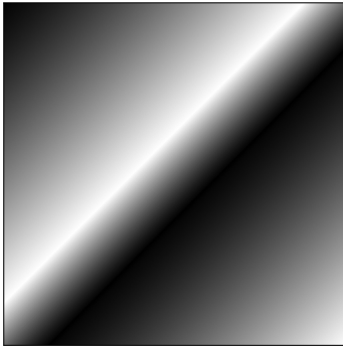


(n) Metoda Bspline3
MSE = 0.0088161
SNR = 6.5659 dB
PSNR = 20.5472 dB
SSIM = 0.76924

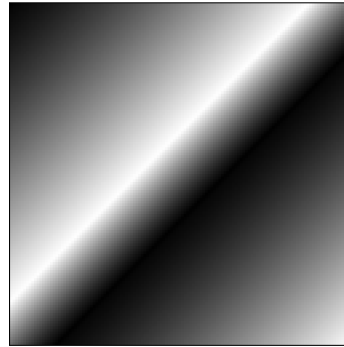


(o) Metoda Bspline4
MSE = 0.0086807
SNR = 6.6331 dB
PSNR = 20.6145 dB
SSIM = 0.77717

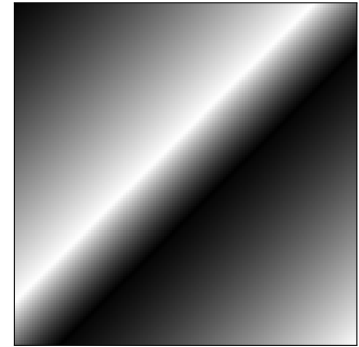
Obr. A.7: Obrázok gradient.png.



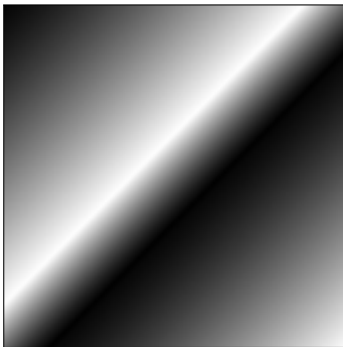
(a) Originálny obrázok



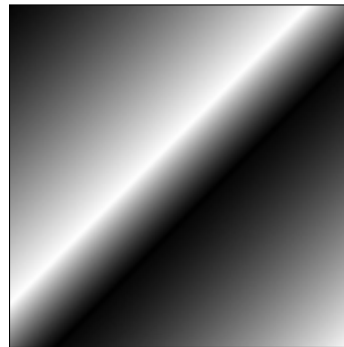
(b) Metoda NN
MSE = 0.000052805
SNR = 31.9818 dB
PSNR = 42.7732 dB
SSIM = 0.98163



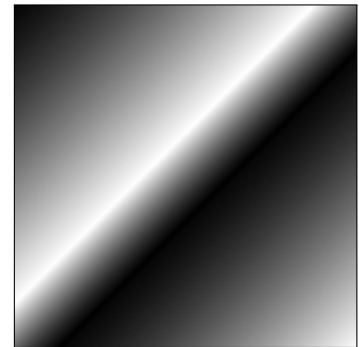
(c) Metoda nearest
MSE = 0.000061738
SNR = 31.3031 dB
PSNR = 42.0945 dB
SSIM = 0.99184



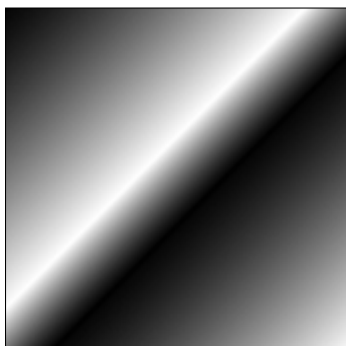
(d) Metoda linear
MSE = 0.000017532
SNR = 36.7702 dB
PSNR = 47.5616 dB
SSIM = 0.99782



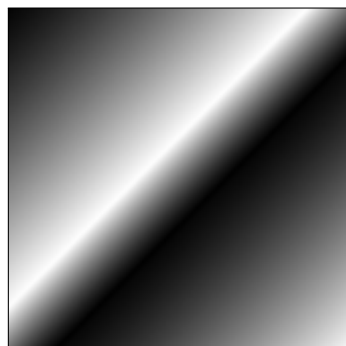
(e) Metoda v5cubic
MSE = 0.000017654
SNR = 36.7403 dB
PSNR = 47.5317 dB
SSIM = 0.99781



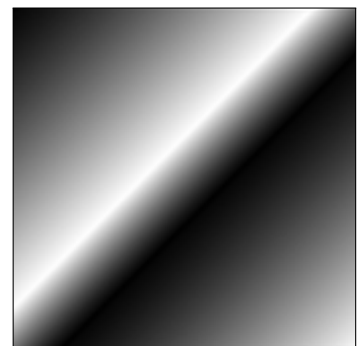
(f) Metoda pchip
MSE = 0.000017567
SNR = 36.7616 dB
PSNR = 47.553 dB
SSIM = 0.99786



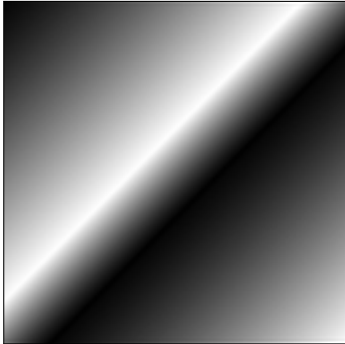
(g) Metoda spline
MSE = 0.000017973
SNR = 36.6624 dB
PSNR = 47.4538 dB
SSIM = 0.99771



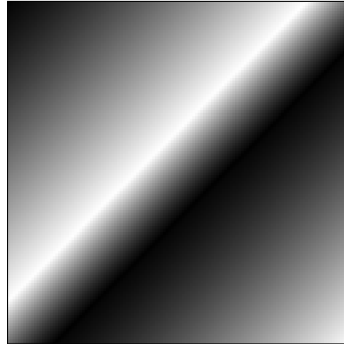
(h) Metoda lineaBspline
MSE = 0.000017532
SNR = 36.7702 dB
PSNR = 47.5616 dB
SSIM = 0.99782



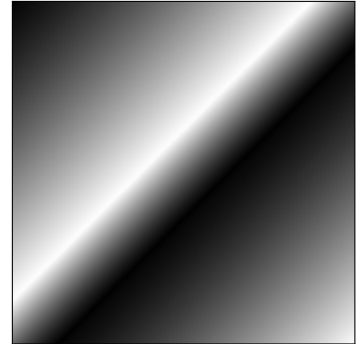
(i) Metoda kvadrBspline
MSE = 0.000031359
SNR = 34.2449 dB
PSNR = 45.0363 dB
SSIM = 0.99782



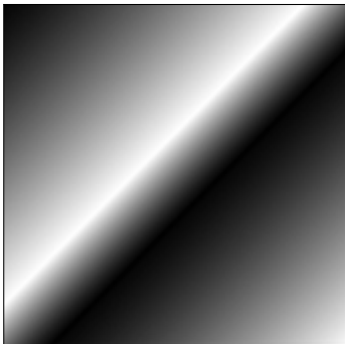
(j) Metoda cubicBspline
MSE = 0.000034251
SNR = 33.8619 dB
PSNR = 44.6533 dB
SSIM = 0.99782



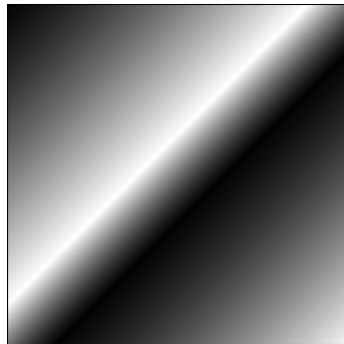
(k) Metoda Bspline0
MSE = 0.000061738
SNR = 31.3031 dB
PSNR = 42.0945 dB
SSIM = 0.99184



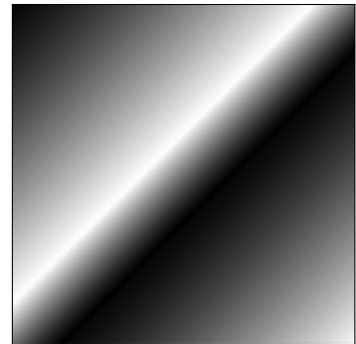
(l) Metoda Bspline1
MSE = 0.000017532
SNR = 36.7702 dB
PSNR = 47.5616 dB
SSIM = 0.99782



(m) Metoda Bspline2
MSE = 0.000025082
SNR = 35.2149 dB
PSNR = 46.0063 dB
SSIM = 0.99769



(n) Metoda Bspline3
MSE = 0.000027709
SNR = 34.7823 dB
PSNR = 45.5737 dB
SSIM = 0.99759



(o) Metoda Bspline4
MSE = 0.000031113
SNR = 34.2792 dB
PSNR = 45.0706 dB
SSIM = 0.99736

Obr. A.8: Obrázok cameraman.png.



(a) Originálny obrázok



(b) Metoda NN
MSE = 0.0027538
SNR = 13.2832 dB
PSNR = 25.6007 dB
SSIM = 0.89494



(c) Metoda nearest
MSE = 0.0031964
SNR = 12.6359 dB
PSNR = 24.9534 dB
SSIM = 0.87943



(d) Metoda linear
MSE = 0.0024618
SNR = 13.7699 dB
PSNR = 26.0874 dB
SSIM = 0.87465



(e) Metoda v5cubic
MSE = 0.0021254
SNR = 14.4081 dB
PSNR = 26.7256 dB
SSIM = 0.89044



(f) Metoda pchip
MSE = 0.0021201
SNR = 14.4189 dB
PSNR = 26.7364 dB
SSIM = 0.89035



(g) Metoda spline
MSE = 0.0020081
SNR = 14.6547 dB
PSNR = 26.9722 dB
SSIM = 0.89543



(h) Metoda lineaBspline
MSE = 0.0024618
SNR = 13.7699 dB
PSNR = 26.0874 dB
SSIM = 0.87465



(i) Metoda kvadrBspline
MSE = 0.0023207
SNR = 14.0262 dB
PSNR = 26.3437 dB
SSIM = 0.87465



(j) Metoda cubicBspline
MSE = 0.0023083
SNR = 14.0495 dB
PSNR = 26.367 dB
SSIM = 0.87465



(k) Metoda Bspline0
MSE = 0.0031964
SNR = 12.6359 dB
PSNR = 24.9534 dB
SSIM = 0.87943



(l) Metoda Bspline1
MSE = 0.0024618
SNR = 13.7699 dB
PSNR = 26.0874 dB
SSIM = 0.87465



(m) Metoda Bspline2
MSE = 0.0020339
SNR = 14.5992 dB
PSNR = 26.9167 dB
SSIM = 0.8946



(n) Metoda Bspline3
MSE = 0.0020054
SNR = 14.6604 dB
PSNR = 26.9779 dB
SSIM = 0.89547



(o) Metoda Bspline4
MSE = 0.0019676
SNR = 14.743 dB
PSNR = 27.0605 dB
SSIM = 0.89699