



**VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ**

BRNO UNIVERSITY OF TECHNOLOGY

**FAKULTA STROJNÍHO INŽENÝRSTVÍ**

FACULTY OF MECHANICAL ENGINEERING

**ÚSTAV MATEMATIKY**

INSTITUTE OF MATHEMATICS

**NUMERICKÉ METODY ZPRACOVÁNÍ OBRAZOVÉ  
INFORMACE PRO REKONSTRUKCI POVRCHU OBJEKTU  
S VYUŽITÍM KONFOKÁLNÍHO MIKROSKOPU**

NUMERICAL METHODS OF IMAGE PROCESSING FOR OBJECT SURFACE RECONSTRUCTION BY  
MEANS OF CONFOCAL MICROSCOPE

**DIPLOMOVÁ PRÁCE**

MASTER'S THESIS

**AUTOR PRÁCE**

AUTHOR

**Bc. Barbora Jedličková**

**VEDOUCÍ PRÁCE**

SUPERVISOR

**doc. Ing. Pavel Štarha, Ph.D.**

**BRNO 2017**

# Zadání diplomové práce

Ústav:	Ústav matematiky
Studentka:	<b>Bc. Barbora Jedličková</b>
Studijní program:	Aplikované vědy v inženýrství
Studijní obor:	Matematické inženýrství
Vedoucí práce:	<b>doc. Ing. Pavel Štarha, Ph.D.</b>
Akademický rok:	2016/17

Ředitel ústavu Vám v souladu se zákonem č.111/1998 o vysokých školách a se Studijním a zkušebním řádem VUT v Brně určuje následující téma diplomové práce:

## **Numerické metody zpracování obrazové informace pro rekonstrukci povrchu objektu s využitím konfokálního mikroskopu**

### **Stručná charakteristika problematiky úkolu:**

Jednou ze základních vlastností konfokálního mikroskopu je získání obrazových informací v dané rovině ostrosti. Při znalosti umístění této roviny můžeme k danému optickému řezu přiřadit jeho výšku. Vzhledem k tomu, že optický řez má nenulovou hloubku ostrosti, může dále za využití záměrné barevné vady objektivu rozlišit rozdíl výšek v rámci jednoho řezu. Tímto získáme potřebné informace pro rekonstrukci povrchu sledovaného objektu.

### **Cíle diplomové práce:**

1. Popsat numerické metody zpracování obrazové informace použité pro rekonstrukci povrchu sledovaného objektu.
2. Určit funkční závislost výšky povrchu objektu na barvě odraženého světla v rámci jednoho optického řezu.
3. Vytvořit aplikační software.

### **Seznam literatury:**

KLÍMA, Miloš, Martin BERNAS, Jiří HOZMAN a Petr DVOŘÁK. Zpracování obrazové informace. Praha: Vydavatelství ČVUT, 1996.

PRATT, William K. Digital Image Processing (Third Edition) PIKS Inside [online]. 3rd ed. New York: Wiley-Interscience, 2001 [cit. 2014-08-07]. ISBN 04-712-2132-5. Dostupné z: <http://www.csupomona.edu/~kding/materials/Digital Image Processing - Third Edition - William K.Pratt.pdf>

Termín odevzdání diplomové práce je stanoven časovým plánem akademického roku 2016/17

V Brně, dne

L. S.

---

prof. RNDr. Josef Šlapal, CSc.  
ředitel ústavu

---

doc. Ing. Jaroslav Katolický, Ph.D.  
děkan fakulty





## **ABSTRAKT**

Tato diplomová práce se zabývá rekonstrukcí povrchu objektu s využitím konfokálního mikroskopu. Práce obsahuje část matematické teorie spojené s tímto problémem. Je zde vysvětlen matematický postup, kterým lze rekonstrukce povrchu objektu provést. Výsledkem práce je aplikační software, kterým je možné tuto rekonstrukci provést. Dále práce obsahuje výsledky pro povrch konkrétního objektu.

## **KLÍČOVÁ SLOVA**

rekonstrukce povrchu, konvoluce, konfokální mikroskop, hyperchromatický objektiv

## **ABSTRACT**

The Diploma thesis deals with object surface reconstruction by means of confocal microscope. It includes part of mathematical theory which is associated with this problem. The mathematical algorithm of the object surface reconstruction is illustrated. This Diploma thesis result is the application development for this reconstruction. The thesis also include the results of specific object.

## **KEYWORDS**

surface reconstruction, convolution, confocal microscope, hyperchromatic lens

JEDLIČKOVÁ, B. *Numerické metody zpracování obrazové informace pro rekonstrukci povrchu objektu s využitím konfokálního mikroskopu*. Brno: Vysoké učení technické v Brně, Fakulta strojního inženýrství, 2017. 50 s. Vedoucí diplomové práce doc. Ing. Pavel Štarha, Ph.D.



## PROHLÁŠENÍ

Prohlašuji, že svou diplomovou práci na téma „Numerické metody zpracování obrazové informace pro rekonstrukci povrchu objektu s využitím konfokálního mikroskopu“ jsem vypracoval(a) samostatně pod vedením vedoucího diplomové práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor(ka) uvedené diplomové práce dále prohlašuji, že v souvislosti s vytvořením této diplomové práce jsem neporušil(a) autorská práva třetích osob, zejména jsem nezasáhl(a) nedovoleným způsobem do cizích autorských práv osobnostních a/nebo majetkových a jsem si plně vědom(a) následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon), ve znění pozdějších předpisů, včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č. 40/2009 Sb.

Brno .....

.....

podpis autorky



## PODĚKOVÁNÍ

Ráda bych na tomto místě poděkovala vedoucímu mé diplomové práce panu doc. Ing. Pavlu Štarhovi, Ph.D. za ochotu, trpělivost a cenné rady a za úžasné konzultace, při kterých mě plnil optimismem, dále pak mojí rodině za podporu při studiu a také mému snoubenci Jiřímu Adámkovi za to, že mi celé studium byl a stále je oporou. DĚKUJI!

Brno .....

.....

podpis autorky



# OBSAH

Úvod	3
<b>1 Matematický aparát</b>	<b>5</b>
1.1 Definice vybraných prostorů	5
1.1.1 Vektorový (lineární) prostor	5
1.1.2 Vektorový prostor se skalárním součinem	7
1.1.3 Afinní (bodový) prostor	8
1.1.4 Euklidovský prostor	10
1.1.5 Projektivní prostor	10
1.2 Numerické metody	15
1.2.1 Aproximace funkcí	15
1.2.2 Numerické derivování	16
1.2.3 Minimalizace funkce jedné proměnné	17
1.2.4 Minimalizace funkce více proměnných	19
1.3 Konvoluce funkce dvou proměnných	23
1.3.1 Diskrétní konvoluce	23
1.3.2 Neúplná konvoluce	24
<b>2 Zpracování obrazové informace</b>	<b>25</b>
2.1 Obraz a jeho vlastnosti	25
2.1.1 Obrazová matice	25
2.1.2 Atributy pixelu	26
2.2 Barevné modely	27
2.2.1 Aditivní model	27
2.2.2 Subtraktivní model	27
2.3 Barevné prostory	28
2.3.1 RGB prostor	28
2.3.2 HSL prostor	28
2.4 Kalibrace digitálních obrazů	29
2.4.1 Základní typy obrazů	29
2.4.2 Metody kalibrace	29
2.5 Histogram	30
2.6 Lineární filtry	30
2.6.1 Základní typy lineárních filtrů	31
<b>3 Rekonstrukce povrchu objektu</b>	<b>35</b>
3.1 Závislost výšky povrchu na barvě	35
3.2 Zpracování obrazu	37
3.3 3D model	40

3.4	Aplikace . . . . .	41
3.4.1	Zjištění závislosti výšky povrchu na barvě . . . . .	41
3.4.2	Zpracování obrazu . . . . .	42
3.4.3	3D modely . . . . .	42
	<b>Závěr</b>	<b>43</b>
	<b>Literatura</b>	<b>45</b>
	<b>Seznam symbolů a zkratk</b>	<b>47</b>
	<b>Seznam příloh</b>	<b>49</b>
	<b>A Příložené CD a 3D modely</b>	<b>50</b>

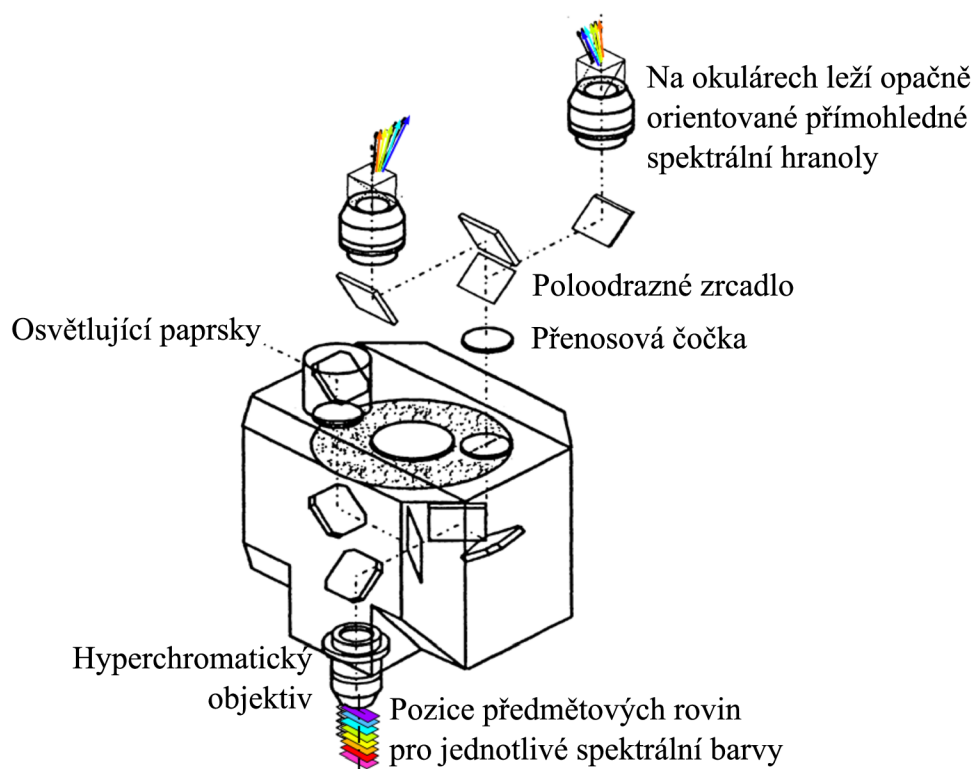


## ÚVOD

Hlavním cílem této diplomové práce byla rekonstrukce povrchu objektu s využitím konfokálního mikroskopu s hyperchromatickým objektivem<sup>1</sup>.

V našem případě se jednalo o konfokální mikroskop s dvojitým řádkováním, jehož princip v roce 1964 vynalezl prof. Mojmír Petráň s doc. Milanem Hadravským.

Konfokální mikroskop je druh optického mikroskopu s vyšší rozlišovací schopností danou detekcí světla pouze z ohniskové roviny mikroskopu. Při pozorování je studovaný vzorek osvětlen bodovým zdrojem světla a speciální konfokální clona skryje zobrazení oblastí nad a pod rovinou zaostření. V případě konfokálního mikroskopu s dvojitým řádkováním je clonou rotující Nipkowovův kotouč (= destička, na které je mnoho vzájemně oddělených clonek), což umožňuje sledovat více bodů preparátu najednou. Hyperchromatický objektiv pak zaručuje, že různé hladiny ostrosti vidíme s různou barvou. Více informací o vynálezu konfokálního mikroskopu s dvojitým řádkováním lze nalézt v [11] a [16].



Obr. 1: Schéma konfokálního mikroskopu s dvojitým řádkováním pro stereoskopii pomocí hyperchromatického objektiv [11]

<sup>1</sup>tj. objektiv s barevnou vadou

K dosažení hlavního cíle bylo třeba splnit další menší cíle. Jedním z nich bylo určení funkční závislosti výšky povrchu objektu na barvě odraženého světla. Metody k tomu potřebné jsou popsány v kapitole 1.2. Jedná se především o metody aproximace funkcí, které následně vedou k metodám vícerozměrné optimalizace a dále k metodám jednorozměrné optimalizace a numerickému derivování.

Dalším cílem bylo vytvořit aplikační software k rekonstrukci povrchu. Ovládání této aplikace je popsáno v kapitole 3.4. Aplikaci jsme se rozhodli vytvořit ve vývojovém prostředí Delphi, vzhledem k jeho vazbě na obor Matematické inženýrství.

Delphi je vývojové prostředí s programovacím jazykem Pascal, s kterým je úzce spjat již od roku 1995, kdy byl poprvé vytvořen firmou Borland. Klíčovým pro tuto firmu byl do roku 1999 (verze Delphi 5), kdy začal být brán jako vedlejší zdroj příjmů. Delphi bylo léta bráno jako zastaralé a „z prachu“ ho vyzvedlo až Embarcadero, které ho odkoupilo v roce 2008. Delphi se pro ně stalo strategickým produktem. Nejnovější verze Delphi mají kompilátory pro jazyk C++, pro Android i iOS.

Pro vytvoření aplikace jsme použili verzi Delphi 10 Seattle. Při vizualizaci zrekonstruovaného povrchu objektu jsme využili knihovnu OpenGL (= Open Graphics Library), s kterou jsme se seznámili v bakalářské práci [5].

# 1 MATEMATICKÝ APARÁT

## 1.1 Definice vybraných prostorů

Pro zobrazení zrekonstruovaného povrchu objektu budeme využívat projektivního prostoru a proto je třeba si jej definovat. K této definici je důležitá znalost pojmů: vektorový prostor, vektorový prostor se skalárním součinem, afinní prostor a euklidovský prostor. Definice těchto pojmů a jejich vlastností jsme čerpali v těchto publikacích: [1], [4], [6], [9] a [10].

### 1.1.1 Vektorový (lineární) prostor

**Definice 1.1.** Necht  $V$  je neprázdná množina, na které je definováno sčítání s vlastnostmi:

(V1)  $\forall \mathbf{u}, \mathbf{v} \in V : \mathbf{u} + \mathbf{v} = \mathbf{w} \in V$  (uzavřenost na množině)

(V2)  $\forall \mathbf{u}, \mathbf{v} \in V : \mathbf{u} + \mathbf{v} = \mathbf{v} + \mathbf{u}$  (komutativita)

(V3)  $\forall \mathbf{u}, \mathbf{v}, \mathbf{w} \in V : \mathbf{u} + (\mathbf{v} + \mathbf{w}) = (\mathbf{u} + \mathbf{v}) + \mathbf{w}$  (asociativita)

(V4)  $\exists \mathbf{o} \in V$  tak, že  $\forall \mathbf{u} \in V : \mathbf{u} + \mathbf{o} = \mathbf{o} + \mathbf{u} = \mathbf{u}$  (existence tzv. nulového prvku)

(V5)  $\forall \mathbf{u} \in V \exists -\mathbf{u} \in V : \mathbf{u} + (-\mathbf{u}) = (-\mathbf{u}) + \mathbf{u} = \mathbf{o}$  (existence tzv. opačného prvku)

a násobení prvků reálnými čísly s vlastnostmi:

(V6)  $\forall \mathbf{u} \in V$  a  $\forall \alpha \in \mathbb{R} : \alpha \cdot \mathbf{u} = \mathbf{w} \in V$  (uzavřenost na množině)

(V7)  $\forall \alpha \in \mathbb{R}$  a  $\forall \mathbf{u}, \mathbf{v} \in V : \alpha \cdot (\mathbf{u} + \mathbf{v}) = \alpha \cdot \mathbf{u} + \alpha \cdot \mathbf{v}$

(V8)  $\forall \alpha, \beta \in \mathbb{R}$  a  $\forall \mathbf{u} \in V : (\alpha + \beta) \cdot \mathbf{u} = \alpha \cdot \mathbf{u} + \beta \cdot \mathbf{u}$

(V9)  $\forall \alpha, \beta \in \mathbb{R}$  a  $\forall \mathbf{u} \in V : \alpha \cdot (\beta \cdot \mathbf{u}) = (\alpha \cdot \beta) \mathbf{u}$

(V10)  $\forall \mathbf{u} \in V : 1 \cdot \mathbf{u} = \mathbf{u}$

Pak  $\mathbb{V} = (V, +)$  nazýváme **vektorový (lineární) prostor (nad  $\mathbb{R}$ )**.

Prvky  $V$  nazýváme *vektory* a značíme je malými tučnými písmeny:  $\mathbf{u}, \mathbf{v}, \dots$ , reálná čísla nazýváme *skaláry*.

Vektor  $\mathbf{u} - \mathbf{v}$  nazýváme *rozdíl vektorů  $\mathbf{u}$  a  $\mathbf{v}$*  a je definován vztahem

$$\mathbf{u} - \mathbf{v} = \mathbf{u} + (-\mathbf{v})$$

*Poznámka.* Místo  $\alpha \cdot \mathbf{u}$  často píšeme pouze  $\alpha \mathbf{u}$ .

*Poznámka.* Axiomy (V7) a (V8) můžeme rozšířit na libovolný počet  $n \in \mathbb{N}$  sčítanců:

$$\alpha(\mathbf{v}_1 + \dots + \mathbf{v}_n) = \alpha \mathbf{v}_1 + \dots + \alpha \mathbf{v}_n$$

$$(\alpha_1 + \dots + \alpha_n) \mathbf{v} = \alpha_1 \mathbf{v} + \dots + \alpha_n \mathbf{v}$$

$\alpha, \alpha_1, \dots, \alpha_n \in \mathbb{R}, \mathbf{v}, \mathbf{v}_1, \dots, \mathbf{v}_n \in V$ .

**Definice 1.2.** Necht  $\mathbb{V} = (V, +)$  je vektorový prostor a  $\mathbf{v}_1, \dots, \mathbf{v}_n \in V$  ( $n \in \mathbb{N}$ ). Řekneme, že vektor  $\mathbf{v} \in V$  je *lineární kombinace vektorů  $\mathbf{v}_1, \dots, \mathbf{v}_n$* , jestliže:

$$\exists \alpha_1, \dots, \alpha_n \in \mathbb{R} : \mathbf{v} = \alpha_1 \mathbf{v}_1 + \dots + \alpha_n \mathbf{v}_n$$

Množinu všech lineárních kombinací vektorů  $\mathbf{v}_1, \dots, \mathbf{v}_n$  značíme  $L(\mathbf{v}_1, \dots, \mathbf{v}_n)$ . Máme tedy

$$L(\mathbf{v}_1, \dots, \mathbf{v}_n) = \{\alpha_1 \mathbf{v}_1 + \dots + \alpha_n \mathbf{v}_n : \alpha_1, \dots, \alpha_n \in \mathbb{R}\}$$

Vektory  $\mathbf{v}_1, \dots, \mathbf{v}_n$  nazýváme *lineárně závislé*, pokud  $\exists \alpha_1, \dots, \alpha_n \in \mathbb{R}$ , z nichž alespoň jedno  $\neq 0$  tak, že platí:

$$\alpha_1 \mathbf{v}_1 + \dots + \alpha_n \mathbf{v}_n = \mathbf{0}$$

V opačném případě vektory  $\mathbf{v}_1, \dots, \mathbf{v}_n$  nazýváme *lineárně nezávislé*.

**Definice 1.3.** Necht  $\mathbb{V} = (V, +)$  je vektorový prostor, necht  $U \subset V$  a platí:

$$\forall \mathbf{u}, \mathbf{v} \in U, \forall \alpha \in \mathbb{R} : \mathbf{u} + \mathbf{v} \in U, \alpha \cdot \mathbf{u} \in U$$

Pak množinu  $U$  nazýváme (*vektorový*) *podprostor* prostoru  $\mathbb{V}$  a píšeme  $\mathbb{U} = (U, +)$ .

**Definice 1.4.** Necht  $\mathbb{V} = (V, +)$  je vektorový prostor a  $\mathbf{v}_1, \dots, \mathbf{v}_n \in V$  ( $n \in \mathbb{N}$ ).

- Soustavu vektorů  $\mathbf{v}_1, \dots, \mathbf{v}_n \in V$  nazýváme *báze vektorového prostoru*  $\mathbb{V}$ , pokud  $\mathbf{v}_1, \dots, \mathbf{v}_n$  jsou lineárně nezávislé a  $\forall \mathbf{v} \in V$  je lineární kombinací vektorů  $\mathbf{v}_1, \dots, \mathbf{v}_n$ . Píšeme  $\mathcal{B} = \langle \mathbf{v}_1, \dots, \mathbf{v}_n \rangle$ .
- Řekneme, že vektory  $\mathbf{v}_1, \dots, \mathbf{v}_n$  tvoří *maximální lineárně nezávislou soustavu vektorů* (prostoru  $\mathbb{V}$ ), pokud jsou vektory  $\mathbf{v}_1, \dots, \mathbf{v}_n$  lineárně nezávislé a  $\forall \mathbf{v} \in V$  jsou vektory  $\mathbf{v}, \mathbf{v}_1, \dots, \mathbf{v}_n$  lineárně závislé.

**Věta 1.1.** Necht  $\mathbb{V} = (V, +)$  je vektorový prostor a  $\mathbf{v}_1, \dots, \mathbf{v}_n \in V$  ( $n \in \mathbb{N}$ ). Pak jsou následující výroky ekvivalentní:

- $\mathcal{B} = \langle \mathbf{v}_1, \dots, \mathbf{v}_n \rangle$ ,  $\mathbf{v}_1, \dots, \mathbf{v}_n \in V$  je báze vektorového prostoru  $\mathbb{V}$ .
- Vektory  $\mathbf{v}_1, \dots, \mathbf{v}_n$  tvoří maximální lineárně nezávislou soustavu vektorů (prostoru  $\mathbb{V}$ ).

*Jsou-li tyto podmínky splněny, pak každý vektor  $\mathbf{v} \in V$  lze jednoznačně zapsat ve tvaru:*

$$\mathbf{v} = \alpha_1 \mathbf{v}_1 + \dots + \alpha_n \mathbf{v}_n$$

$$\alpha_1, \dots, \alpha_n \in \mathbb{R}$$

Čísla  $\alpha_1, \dots, \alpha_n \in \mathbb{R}$  nazýváme *souřadnice vektoru  $\mathbf{v}$  vzhledem k bázi  $\mathcal{B} = \langle \mathbf{v}_1, \dots, \mathbf{v}_n \rangle$*  a píšeme  $\mathbf{v} = (\alpha_1, \dots, \alpha_n)_{\mathcal{B}}$ .

*Důkaz.* viz [6] □

**Definice 1.5.** Necht  $\mathbb{V} = (V, +)$  je vektorový prostor a  $\mathcal{B} = \langle \mathbf{v}_1, \dots, \mathbf{v}_n \rangle$  je jeho báze. Pak číslo  $n$  nazýváme *dimenze vektorového prostoru*  $\mathbb{V}$  a píšeme:

$$\dim \mathbb{V} = n$$

### 1.1.2 Vektorový prostor se skalárním součinem

**Definice 1.6.** Necht  $\mathbb{V} = (V, +)$  je vektorový prostor. Necht  $\langle \cdot, \cdot \rangle : V \times V \rightarrow \mathbb{R}$  je zobrazení splňující:

$$(VS1) \quad \forall \mathbf{u}, \mathbf{v} \in V : \langle \mathbf{u}, \mathbf{v} \rangle = \langle \mathbf{v}, \mathbf{u} \rangle$$

$$(VS2) \quad \forall \mathbf{u}, \mathbf{v}, \mathbf{w} \in V : \langle \mathbf{u} + \mathbf{v}, \mathbf{w} \rangle = \langle \mathbf{u}, \mathbf{w} \rangle + \langle \mathbf{v}, \mathbf{w} \rangle$$

$$(VS3) \quad \forall \mathbf{u}, \mathbf{v} \in V, \forall \alpha \in \mathbb{R} : \langle \alpha \mathbf{u}, \mathbf{v} \rangle = \alpha \langle \mathbf{u}, \mathbf{v} \rangle$$

$$(VS4) \quad \forall \mathbf{u} \in V : \langle \mathbf{u}, \mathbf{u} \rangle \geq 0$$

$$(VS5) \quad \forall \mathbf{u} \in V : \langle \mathbf{u}, \mathbf{u} \rangle = 0 \Leftrightarrow \mathbf{u} = \mathbf{o}$$

Pak toto zobrazení nazýváme *skalární součin* a prostor  $\mathbb{V}$  nazýváme **vektorový prostor se skalárním součinem**.

*Příklad.* Pro  $\mathbf{u} = (u_1, \dots, u_n), \mathbf{v} = (v_1, \dots, v_n) \in \mathbb{R}^n, n \in \mathbb{N}$  mějme

$$\langle \mathbf{u}, \mathbf{v} \rangle = u_1 v_1 + u_2 v_2 + \dots + u_n v_n. \quad (1.1)$$

Pak zobrazení  $\langle \cdot, \cdot \rangle : \mathbb{R}^n \times \mathbb{R}^n \rightarrow \mathbb{R}$  splňuje axiomy (VS1) - (VS4). Vektorový prostor  $\mathbb{R}^n$  je tedy vektorový prostor se skalárním součinem.

Skalární součin  $\langle \mathbf{u}, \mathbf{v} \rangle$  v tomto případě značíme  $\mathbf{u} \cdot \mathbf{v}$  a nazýváme *bodový součin*.

**Definice 1.7.** Necht  $(V, +)$  je euklidovský prostor. Pro  $\mathbf{v} \in V$  položme

$$\|\mathbf{v}\| = \sqrt{\langle \mathbf{v}, \mathbf{v} \rangle}.$$

Číslo  $\|\mathbf{v}\|$  nazýváme *norma* nebo *délka* nebo *velikost vektoru*  $\mathbf{v}$ . Dle axiomů (VS4) a (VS5) je  $\|\mathbf{v}\| > 0$  pro  $\mathbf{v} \neq \mathbf{o}$  a  $\|\mathbf{o}\| = 0$ .

V případě, že  $\|\mathbf{v}\| = 1$ , pak vektor  $\mathbf{v} \in V$  nazýváme *jednotkový vektor*.

Operaci, kdy vektor  $\mathbf{v}$  násobíme jeho vlastní normou, tj.

$$\frac{1}{\|\mathbf{v}\|} \mathbf{v}$$

nazýváme *normování vektoru*.

*Příklad.* Pro  $\mathbf{v} = (v_1, \dots, v_n) \in \mathbb{R}^n$  platí

$$\|\mathbf{v}\| = \sqrt{v_1^2 + \dots + v_n^2}. \quad (1.2)$$

**Věta 1.2** (Schwarzova nerovnost). *Necht  $\mathbb{V} = (V, +)$  je vektorový prostor se skalárním součinem a  $\mathbf{u}, \mathbf{v} \in V$  pak platí*

$$\langle \mathbf{u}, \mathbf{v} \rangle^2 \leq \|\mathbf{u}\|^2 \cdot \|\mathbf{v}\|^2 \quad (1.3)$$

*Rovnost nastane právě, když jeden z vektorů  $\mathbf{u}, \mathbf{v}$  je násobkem druhého.*

*Důkaz.*

$$\mathbf{u} = \mathbf{o}: \langle \mathbf{o}, \mathbf{v} \rangle \stackrel{(VS2)}{=} \langle \mathbf{o}, \mathbf{v} \rangle + \langle \mathbf{o}, \mathbf{v} \rangle \Rightarrow \langle \mathbf{o}, \mathbf{v} \rangle = 0 \Rightarrow 0 \leq 0 \cdot \|\mathbf{v}\|^2$$

Stejně tak i pro  $\mathbf{v} = \mathbf{o}$  a pro  $\mathbf{u}, \mathbf{v} = \mathbf{o}$ .

$$\mathbf{u}, \mathbf{v} \neq \mathbf{o}: \text{Nechť } \hat{\mathbf{u}} = \frac{1}{\|\mathbf{u}\|} \mathbf{u} \text{ a } \hat{\mathbf{v}} = \frac{1}{\|\mathbf{v}\|} \mathbf{v}$$

$$0 \leq \|\hat{\mathbf{u}} - \hat{\mathbf{v}}\|^2 = \langle \hat{\mathbf{u}} - \hat{\mathbf{v}}, \hat{\mathbf{u}} - \hat{\mathbf{v}} \rangle = \langle \hat{\mathbf{u}}, \hat{\mathbf{u}} \rangle - \langle \hat{\mathbf{u}}, \hat{\mathbf{v}} \rangle - \langle \hat{\mathbf{v}}, \hat{\mathbf{u}} \rangle + \langle \hat{\mathbf{v}}, \hat{\mathbf{v}} \rangle = \|\hat{\mathbf{u}}\|^2 - 2\langle \hat{\mathbf{u}}, \hat{\mathbf{v}} \rangle + \|\hat{\mathbf{v}}\|^2 = 2(1 - \langle \hat{\mathbf{u}}, \hat{\mathbf{v}} \rangle) \Rightarrow \langle \hat{\mathbf{u}}, \hat{\mathbf{v}} \rangle \leq 1.$$

$$\text{Podobně pro } 0 \leq \|\hat{\mathbf{u}} + \hat{\mathbf{v}}\|^2 = 2(1 + \langle \hat{\mathbf{u}}, \hat{\mathbf{v}} \rangle) \Rightarrow -1 \leq \langle \hat{\mathbf{u}}, \hat{\mathbf{v}} \rangle.$$

$$\text{Jelikož } \langle \hat{\mathbf{u}}, \hat{\mathbf{v}} \rangle = \frac{1}{\|\mathbf{u}\| \cdot \|\mathbf{v}\|} \langle \mathbf{u}, \mathbf{v} \rangle, \text{ potom } -1 \leq \frac{1}{\|\mathbf{u}\| \cdot \|\mathbf{v}\|} \langle \mathbf{u}, \mathbf{v} \rangle \leq 1 \Rightarrow \langle \mathbf{u}, \mathbf{v} \rangle^2 \leq \|\mathbf{u}\|^2 \cdot \|\mathbf{v}\|^2$$

□

**Definice 1.8.** Necht  $\mathbb{V} = (V, +)$  je vektorový součin se skalárním součinem.

(a) Jsou-li vektory  $\mathbf{o} \neq \mathbf{u}, \mathbf{v} \in V$ , pak dle Věty(1.2) platí:

$$-1 \leq \frac{\langle \mathbf{u}, \mathbf{v} \rangle}{\|\mathbf{u}\| \cdot \|\mathbf{v}\|} \leq 1$$

$\exists! \varphi \in \mathbb{R}, 0 \leq \varphi \leq \pi$  tak, že  $\cos \varphi = \frac{\langle \mathbf{u}, \mathbf{v} \rangle}{\|\mathbf{u}\| \cdot \|\mathbf{v}\|}$ .  $\varphi$  se pak nazývá *úhel vektorů  $\mathbf{u}, \mathbf{v}$*

(b) Pokud pro vektory  $\mathbf{u}, \mathbf{v} \in V$  platí:

$$\langle \mathbf{u}, \mathbf{v} \rangle = 0$$

pak tyto vektory nazýváme *ortogonální (kolmé)*.

(c) Necht  $\{\mathbf{v}_1, \dots, \mathbf{v}_n \in V\}$ ,  $n \in \mathbb{N}$  je množina vektorů, pro kterou platí:

$$(i) \forall i \in \mathbb{N}, 1 \leq i \leq n: \mathbf{v}_i \neq \mathbf{o}$$

$$(ii) \forall i, j \in \mathbb{N}, 1 \leq i, j \leq n, i \neq j: \langle \mathbf{v}_i, \mathbf{v}_j \rangle = 0$$

Pak tuto množinu nazýváme *ortogonální množina vektorů*.

(d) Necht  $\{\mathbf{v}_1, \dots, \mathbf{v}_n \in V\}$ ,  $n \in \mathbb{N}$  je ortogonální množina vektorů, pro kterou platí:

$$\forall i \in \mathbb{N}, 1 \leq i \leq n: \|\mathbf{v}_i\| = 1$$

Pak tuto množinu nazýváme *ortonormální množina vektorů*.

**Definice 1.9.** Necht  $\mathbb{V} = (V, +)$  je vektorový prostor se skalárním součinem a  $\mathcal{B} = \{\mathbf{v}_1, \dots, \mathbf{v}_n\}$ ,  $n \in \mathbb{N}$  je jeho báze. Pokud je množina  $\{\mathbf{v}_1, \dots, \mathbf{v}_n\}$  ortogonální, resp. ortonormální, nazýváme bázi  $\mathcal{B}$  *ortogonální, resp. ortonormální báze*.

### 1.1.3 Afinní (bodový) prostor

**Definice 1.10.** Necht  $\mathbb{V} = (V, +)$  je vektorový prostor,  $\mathcal{A}$  je neprázdná množina a zobrazení  $g: \mathcal{A} \times \mathcal{A} \rightarrow V$  splňuje:

$$(A1) \forall A \in \mathcal{A} \text{ a } \forall \mathbf{u} \in V: \exists! B \in \mathcal{A}: (A, B) = \mathbf{u}$$

$$(A2) \forall A, B, C \in \mathcal{A}: g(A, C) = g(A, B) + g(B, C)$$

Pak množinu  $\mathcal{A}$  nazýváme **afinní (bodový) prostor**.

Prvky množiny  $\mathcal{A}$  nazýváme *body*. Vektorový prostor  $\mathbb{V}$  nazýváme *zaměření* prostoru  $\mathcal{A}$ .

Je-li  $\dim \mathbb{V} = n$ , pak říkáme, že afinní prostor  $\mathcal{A}$  je  $n$ -rozměrný a píšeme  $\dim \mathcal{A} = n$ , příp.  $\mathcal{A}_n$ . Prostor  $\mathcal{A}_1$  nazýváme *afinní přímka*,  $\mathcal{A}_2$  *afinní rovina* a  $\mathcal{A}_3$  *afinní prostor*.

*Poznámka.*  $(A, B) \equiv \overrightarrow{AB} \equiv B - A$

**Definice 1.11.** Necht  $\mathcal{A}$  je afinní prostor se zaměřením  $\mathbb{V} = (V, +)$ , necht  $\mathcal{B} \subset \mathcal{A}$ ,  $\mathcal{B} \neq \emptyset$  a necht  $U = \{g(A, B) : A, B \in \mathcal{B}\}$ . Pokud platí:

- (1)  $\mathbb{U} = (U, +)$ , tzn. je vektorovým podprostorem  $\mathbb{V}$ ,
- (2)  $\forall A \in \mathcal{B}, \forall \mathbf{u} \in U \exists B \in \mathcal{B} : (A, B) = \mathbf{u}$ .

Pak  $\mathcal{B}$  nazýváme *afinní podprostor*  $\mathcal{A}$  se změřením  $\mathbb{U}$ .

**Definice 1.12.** Necht  $\mathcal{B}$  je podprostor afinního prostoru  $\mathcal{A}$ . Je-li

$$\dim \mathcal{B} = \begin{cases} 0 & \mathcal{B} \text{ nazýváme } \textit{bod} \text{ prostoru } \mathcal{A} \\ 1 & \mathcal{B} \text{ nazýváme } \textit{přímka} \text{ prostoru } \mathcal{A} \\ 2 & \mathcal{B} \text{ nazýváme } \textit{rovina} \text{ prostoru } \mathcal{A} \\ \dim \mathcal{A} - 1 & \mathcal{B} \text{ nazýváme } \textit{nadrovina} \text{ prostoru } \mathcal{A} \end{cases}$$

**Definice 1.13.** Necht  $\mathcal{A}$  je afinní prostor,  $\dim \mathcal{A} \geq 1$ , necht  $\mathcal{B} = \langle \mathbf{v}_1, \dots, \mathbf{v}_n \rangle$  je báze zaměření tohoto prostoru a necht  $O \in \mathcal{A}$  je pevný bod. Pak systém

$$\mathcal{R} = \langle O, \mathbf{v}_1, \dots, \mathbf{v}_n \rangle$$

nazýváme *afinní repér (afinní souřadnicový systém)*. Bod  $O$  nazýváme *počátek* a vektory  $\mathbf{v}_1, \dots, \mathbf{v}_n$  *základní vektory* afinního repéru  $\mathcal{R}$ .

Přímky  $x_i = \{O, L(\mathbf{v}_i)\}$  nazýváme *osy afinních souřadnic (souřadné osy afinního repéru)*.

**Definice 1.14.** Necht  $\mathcal{A}$  je afinní prostor se zaměřením  $\mathbb{V} = (V, +)$  a repérem  $\mathcal{R} = \langle O, \mathbf{v}_1, \dots, \mathbf{v}_n \rangle$ ,  $A \in \mathcal{A}$  je bod a  $\mathbf{u} \in V$  je vektor.

Uspořádaná  $n$ -tice  $[a_1, \dots, a_n]$ ,  $a_i \in \mathbb{R} \ i = 1, \dots, n$  splňující:

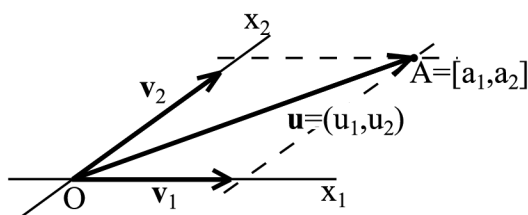
$$g(O, B) = a_1 \mathbf{v}_1 + \dots + a_n \mathbf{v}_n$$

nazýváme *souřadnice bodu*  $A$  v afinním repéru  $\mathcal{R}$ .

Uspořádaná  $n$ -tice  $(u_1, \dots, u_n)$ ,  $u_i \in \mathbb{R}, \ i = 1, \dots, n$  splňující:

$$\mathbf{u} = u_1 \mathbf{v}_1 + \dots + u_n \mathbf{v}_n$$

nazýváme *souřadnice vektoru*  $\mathbf{u}$ .



Obr. 1.1: Příklad afinního repéru [4]

### 1.1.4 Euklidovský prostor

**Definice 1.15.** Necht  $\mathcal{A}$  je afinní prostor,  $\mathbb{V} = (V, +)$  je vektorový prostor se skalárním součinem a platí, že  $\mathbb{V}$  je zaměření afinního prostoru  $\mathcal{A}$ .

Pak tento afinní prostor nazýváme **euklidovský prostor**.

$n$ -rozměrný euklidovský prostor značíme  $\mathbb{E}_n$ .

**Definice 1.16.** Necht  $\mathbb{E}_n$ ,  $n \in \mathbb{N}$  je  $n$ -rozměrný euklidovský prostor a necht  $\mathcal{R} = \langle O, \mathbf{v}_1, \dots, \mathbf{v}_n \rangle$  je jeho repér takový, že  $\mathcal{B} = \langle \mathbf{v}_1, \dots, \mathbf{v}_n \rangle$  je ortonormální báze zaměření tohoto prostoru.

Pak repér  $\mathcal{R}$  nazýváme *kartézský repér (ortonormální repér)* nebo *kartézský souřadný systém v  $\mathbb{E}_n$* .

Souřadnice bodů v  $\mathbb{E}_n$  vzhledem k repéru  $\mathcal{R}$  nazýváme *kartézské souřadnice*.

### 1.1.5 Projektivní prostor

**Definice 1.17.** Necht  $\mathbb{V} = (V, +)$  je vektorový prostor,  $\dim \mathbb{V} = n+1$ ,  $n \geq -1$ , necht  $\mathbb{P}_n$  je množina všech podprostorů  $U_i$ ,  $i = 1, \dots, n$  prostoru  $\mathbb{V}$  tak, že  $\forall i \in \{1, \dots, n\} : \dim U_i = 1$ . Pak množinu  $\mathbb{P}_n$  nazýváme  **$n$ -rozměrný projektivní prostor**.

Vektorový prostor  $\mathbb{V}$ , pak nazýváme *aritmetický základ projektivního prostoru  $\mathbb{P}_n$* , jeho bázi, pak *báze projektivního prostoru  $\mathbb{P}_n$*

Každý vektor  $\mathbf{o} \neq \mathbf{p} \in V$  *aritmetický bod projektivního prostoru  $\mathbb{P}_n$* .

Množinu  $\bar{A} = \{\mathbf{u} \in \mathbb{P}_n : \mathbf{u} = k\mathbf{p}, k \neq 0\}$  nazýváme *geometrický bod projektivního prostoru  $\mathbb{P}_n$*  a složky vektoru  $\mathbf{u} = (kp_1, kp_2, \dots, kp_{n+1})$  *homogenní souřadnice bodu  $\bar{A}$*  a píšeme:  $\bar{A} = (kp_1, kp_2, \dots, kp_{n+1})$ .

Typickým modelem projektivního prostoru je euklidovský prostor rozšířený o jednu dimenzi, tzv. **rozšířený euklidovský prostor**. Budeme ho značit  $\bar{\mathbb{E}}_n$ , přičemž  $\dim \bar{\mathbb{E}}_n = n+1$ .

**Definice 1.18.** Necht  $\bar{A} = (a_1^*, \dots, a_{n+1}^*)$  je bod projektivního prostoru.

- (1) Pokud  $a_{n+1}^* \neq 0$ , pak tento bod nazýváme *vlastní* a můžeme ho reprezentovat jako  $\bar{A} = \left(\frac{a_1^*}{a_{n+1}^*}, \dots, \frac{a_n^*}{a_{n+1}^*}, 1\right) = (a_1, \dots, a_n, 1)$ .
- (2) Pokud  $a_{n+1}^* = 0$ , pak tento bod nazýváme *nevlastní* a značíme ho  $\bar{A}_\infty = (a_1, \dots, a_n, 0)$



*Poznámka.* Projektivní rovinu (prostor) můžeme chápat jako euklidovskou rovinu (prostor) rozšířenou (rozšířený) o nevlastní body... Vlastní body projektivního prostoru lze chápat jako euklidovské..., nevlastní body chápeme jako směry, jejich reprezentanty jako vektory daného směru. ([9], s. 38)

**Definice 1.19.** Necht  $\mathbb{P}_n$  je projektivní prostor s aritmetickým základem  $\mathbb{V} = (V, +)$ ,  $\mathbb{P}_m$  je projektivní prostor s aritmetickým základem  $\mathbb{U} = (U, +)$ , necht  $f : \mathbb{P}_m \rightarrow \mathbb{P}_n$  je zobrazení. Pokud existuje zobrazení  $g : \mathbb{V} \rightarrow \mathbb{U}$  takové, že  $\forall \mathbf{v} \in V$  a  $\forall \bar{X} \in \mathbb{P}_n$  platí, že složky vektoru  $g(\mathbf{v}) \in U$  jsou homogenní souřadnice bodu  $f(\bar{X}) \in \mathbb{P}_m$ , pak zobrazení  $f$  nazýváme *projektivní zobrazení*.

*Poznámka.* Projektivní zobrazení bodu  $\bar{X} = (x_1, \dots, x_{n+1})$  na bod  $\bar{X}' = (x'_1, \dots, x'_{m+1})$  lze popsat jako soustavu rovnic:

$$f(\bar{X}) = \mathbf{A} \cdot \bar{X}$$

Tedy:

$$\begin{pmatrix} x'_1 \\ x'_2 \\ \vdots \\ x'_{m+1} \end{pmatrix} = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n+1} \\ a_{21} & \cdots & \cdots & a_{2n+1} \\ \vdots & \ddots & \ddots & \vdots \\ a_{m+11} & \cdots & \cdots & a_{m+1n+1} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_{n+1} \end{pmatrix}$$

Maticově:

$$\bar{X}'^T = \mathbf{A} \cdot \bar{X}^T$$

kde  $\mathbf{A}$  je matice určující zobrazení.

Výhodou maticové reprezentace těchto zobrazení je, že je lze snadno skládat a to prostým vynásobením všech matic námi požadovaných zobrazení.

### Zobrazení v projektivním prostoru $\bar{\mathbb{E}}_3$

Pro názornost si představíme některá zobrazení v projektivním prostoru.

Bavíme se tedy o zobrazením bodu  $\bar{X} = (x_1, x_2, x_3, x_4)$  na bod  $\bar{X}' = (x'_1, x'_2, x'_3, x'_4)$ :

$$\begin{pmatrix} x'_1 \\ x'_2 \\ x'_3 \\ x'_4 \end{pmatrix} = \begin{pmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix}$$

Obr. 1.2: Vybraná zobrazení v projektivním prostoru [9]

Název zobrazení	Matice zobrazení
Souměrnost podle roviny $z = 0$	$\mathbf{S}_{xy} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$
Souměrnost podle osy $y$	$\mathbf{S}_y = \begin{pmatrix} -1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$
Souměrnost podle počátku	$\mathbf{S} = \begin{pmatrix} -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$
Posunutí o vektor $\mathbf{v}$	$\mathbf{T}_v = \begin{pmatrix} 1 & 0 & 0 & v_1 \\ 0 & 1 & 0 & v_2 \\ 0 & 0 & 1 & v_3 \\ 0 & 0 & 0 & 1 \end{pmatrix}$
Otočení kolem osy $x$ o úhel $\alpha$	$\mathbf{R}_{x,\alpha} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos \alpha & -\sin \alpha & 0 \\ 0 & \sin \alpha & \cos \alpha & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$
Otočení kolem osy $z$ o úhel $\alpha$	$\mathbf{R}_{z,\alpha} = \begin{pmatrix} \cos \alpha & -\sin \alpha & 0 & 0 \\ \sin \alpha & \cos \alpha & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$

### Promítání (= zobrazení) prostoru na rovinu

**Definice 1.20.** Necht  $\bar{\mathbb{E}}_3$  je projektivní prostor, necht  $\pi = \bar{\mathbb{E}}_2$  je projektivní rovina a necht  $\bar{S} \notin \pi$ . Zobrazení  $\mathcal{P} : \bar{\mathbb{E}}_3 \setminus \{\bar{S}\} \rightarrow \pi$  splňující:

$$\forall \bar{X} \in \bar{\mathbb{E}}_3 : \bar{X} \neq \bar{S} \mapsto \bar{X}' \in (\bar{S}\bar{X} \cap \pi)$$

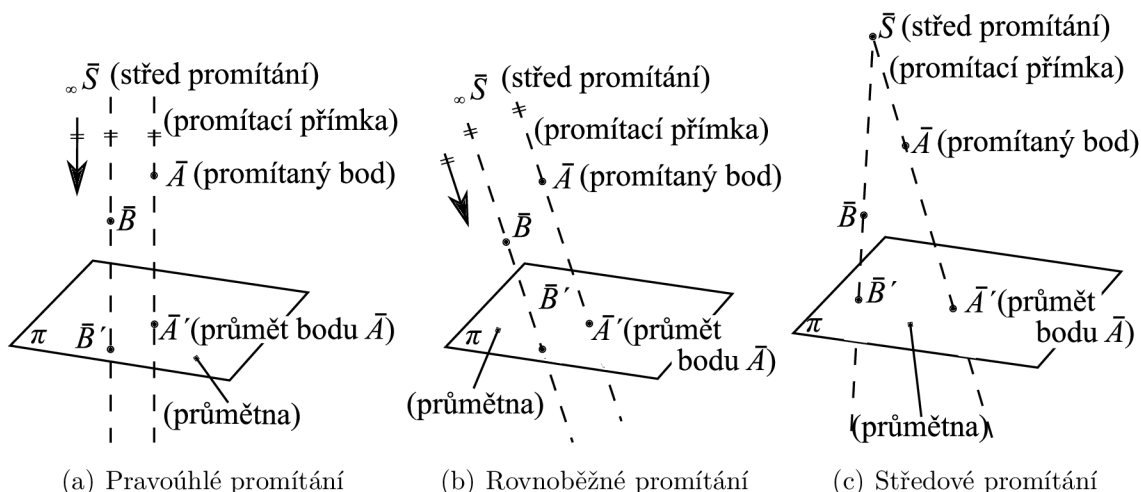
nazýváme *promítání*.

Bod  $\bar{S}$  nazýváme *střed promítání*, rovinu  $\pi$  *průmětna*, přímku  $\bar{S}\bar{X}$  *promítací přímka*.

V případě, že bod  $\bar{S}$  je

- vlastní, pak promítání nazýváme *středové*
- nevlastní, pak promítání nazýváme *rovnoběžné*

Rovnoběžné promítání, které má promítací přímky kolmé na průmětnu, nazýváme *pravoúhlé*.



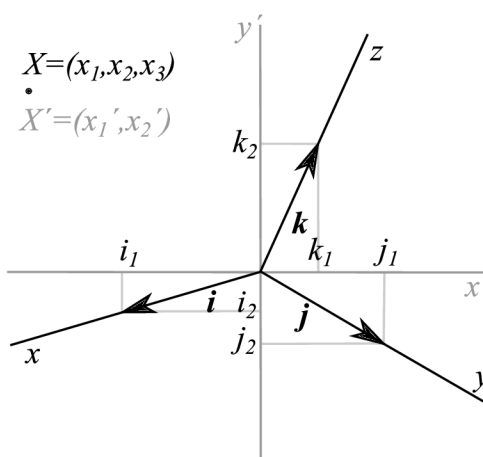
Obr. 1.3: Promítání prostoru na rovinu [9]

Vzhledem k tomu, že chceme simulovat reálný pohled na předmět, je pro nás vhodné používat **pravoúhlé promítání**.

Maticově lze pravoúhlé promítání bodu  $\bar{X} = (x_1, x_2, x_3, 1)$  na bod  $\bar{X}' = (x'_1, x'_2, 1)$  zapsat takto:

$$\begin{pmatrix} x'_1 \\ x'_2 \\ 1 \end{pmatrix} = \begin{pmatrix} i_1 & j_1 & k_1 & 0 \\ i_2 & j_2 & k_2 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ 1 \end{pmatrix}$$

kde  $\mathbf{i} = (i_1, i_2)$ ,  $\mathbf{j} = (j_1, j_2)$ ,  $\mathbf{k} = (k_1, k_2)$  jsou báze vektory daného  $\bar{\mathbb{E}}_3$  vzhledem k rovině  $\pi$  viz obrázek 1.4.

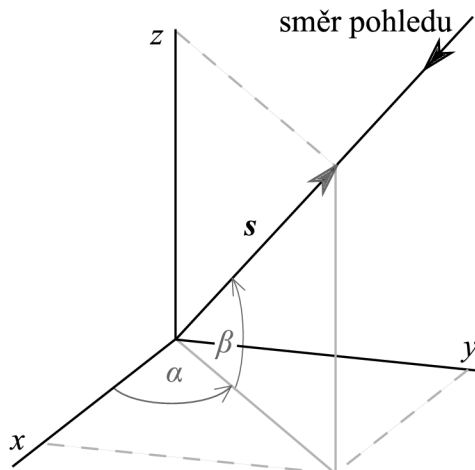


Obr. 1.4: Pravoúhlé promítání prostoru na rovinu [10]

*Příklad* (Pravoúhlé promítání na rovinu  $z = 0$ ). V tomto případě pro matici určující zobrazení platí:

$$\mathbf{A} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Pokud bychom chtěli pravoúhle promítat na obecnou rovinu, je třeba znát rovnici této roviny nebo znát směr pohledu pomocí horizontálního úhlu  $\alpha$  a vertikálního úhlu  $\beta$  (viz obrázek 1.5). Směrový vektor pohledu  $\mathbf{s} = (s_1, s_2, s_3)$  je jednotkový vektor se souřadnicemi:



Obr. 1.5: Směr pohledu [10]

$$\begin{aligned} s_1 &= \cos \alpha \cos \beta \\ s_2 &= \sin \alpha \cos \beta \\ s_3 &= \sin \beta \end{aligned} \tag{1.4}$$

Pozice pozorovatele je bod  $\bar{P} = (ds_1, ds_2, ds_3)$ , kde  $d$  je vzdálenost pozorovatele od počátku souřadné soustavy.

Matice pravoúhlého promítání na obecnou rovinu lze odvodit následovně:

Pomocí dvou rotací ztotožníme směr pohledu s některou souřadnou osou a poté provedeme kolmý průmět do příslušné průmětny. Matice tohoto zobrazení má tvar:

$$\mathbf{A} = \mathbf{K}_z \cdot \mathbf{R}_{x, \beta - \frac{\pi}{2}} \cdot \mathbf{R}_{z, -\alpha - \frac{\pi}{2}}$$

V praxi se často využívá i **středové promítání** (např. lineární perspektiva, ve fotogrametrii). Jak již bylo uvedeno, toto promítání má vlastní střed promítání.

Matice středového promítání do roviny  $z = 0$ , přičemž střed promítání leží na ose  $z$ :  $\bar{S} = (s_1, s_2, s_3, 1)$  má tvar:

$$\mathbf{P}_{\bar{S}, xy} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & -\frac{1}{s_3} & 1 \end{pmatrix}$$

Maticе středového promítání na obecnou rovinu lze odvodit analogicky jako u pravoúhlého promítání:

$$\mathbf{A} = \mathbf{P}_{\bar{S},xy} \cdot \mathbf{R}_{x,\beta-\frac{\pi}{2}} \cdot \mathbf{R}_{z,-\alpha-\frac{\pi}{2}}$$

## 1.2 Numerické metody

Tato kapitola se věnuje důležitým numerickým metodám, použitých pro řešení této diplomové práce. Popis těchto metod byl čerpán z publikací [3],[7] [12] a [15].

Numerické metody jsou důležitou matematickou disciplínou, neboť nám umožňují řešit netriviální úlohy v oblasti řešení soustav lineárních rovnic, aproximace funkcí, numerického derivování a jednorozměrné i vícerozměrné optimalizace. Numerickou metodu realizují kroky, jejichž popis nazýváme algoritmus numerické metody.

### 1.2.1 Aproximace funkcí

Aproximace funkce  $f(x)$  znamená její nahrazení funkcí  $\varphi(x)$ , která je jí v jistém smyslu blízká. Píšeme  $\varphi(x) \approx f(x)$ . Základními typy aproximace jsou různé způsoby *interpolací* a *metody prokládání dat* (např. *metoda nejmenších čtverců*).

Interpolace je taková aproximace, při které funkce  $\varphi(x)$  nabývá v zadaných bodech  $x_i$  předepsaných hodnot  $y_i = f(x_i)$ . Nejčastěji se jedná o polynom, příp. po částech polynom.

Naproti tomu metoda prokládání dat je taková aproximace, kdy funkce  $\varphi(x)$  nemusí nabývat v zadaných bodech  $x_i$  předepsaných hodnot  $y_i = f(x_i)$ , ale „vzdálenost“ mezi funkcí  $\varphi(x)$  a body  $[x_i, y_i]$  by měla být v jistém smyslu minimální.

#### Metoda nejmenších čtverců (MNČ)

Jedná se o neznámější metodu prokládání dat, která je tvaru:

Nechť  $x$  je nezávisle proměnná a  $y(x)$  je neznámá funkce proměnné  $x$ , kterou chceme aproximovat funkcí  $\varphi(x)$ . Nechť jsme provedli  $m$  pozorování:

$$y_i = y(x_i), \quad i = 1, \dots, m$$

Rezidua<sup>1</sup> mají tvar:

$$r_i = y_i - \varphi(x_i)$$

MNČ řeší úlohu na základě minimalizace kvadrátů jejich reziduí:

$$\|\mathbf{r}\|^2 = \sum_i r_i^2 = \sum_i (y_i - \varphi(x_i))^2 \rightarrow \min \quad (1.5)$$

kde  $\mathbf{r} = (r_1, r_2, \dots)$  je vektor reziduí.

Což se dále řeší metodami popsány v kapitole 1.2.4.

---

<sup>1</sup>= rozdíl mezi pozorováním a modelem

Řešení musí splňovat nutnou podmínku pro extrém:

$$\frac{\partial \|\mathbf{r}\|^2}{\partial x_k} = \frac{\partial}{\partial x_k} \sum_i r_i^2 = 0, \quad k = 1, 2, \dots, n$$

## 1.2.2 Numerické derivování

Použití numerického derivování je vhodné například v případě, je-li funkce  $f(x)$ , kterou chceme derivovat, natolik složitá, že výpočet její derivace je příliš pracný.

Jedná se vlastně o aproximaci  $f^{(k)}(x) \approx \varphi^{(k)}(x)$ , kde  $k$  je řád derivace. Vhodným způsobem je aproximace derivací Lagrangeova interpolačního polynomu  $f^{(k)}(x) \approx P_n^{(k)}(x)$ .

### Chyba aproximace v uzlovém bodě

Nechť  $f \in C^{n+1}\langle a, b \rangle^2$ , kde  $a$  je nejmenší a  $b$  je největší z uzlů interpolace. Pak pro chybu  $f'(x_s) - P_n'(x_s)$  v některém z uzlů  $x_s$  platí:

$$f'(x_s) - P_n'(x_s) = \frac{f^{(n+1)}(\xi_s)}{(n+1)!} \omega'_{n+1}(x_s), \quad (1.6)$$

kde  $\omega_{n+1}(x_s) = (x - x_0)(x - x_1) \cdots (x - x_n)$  a  $\xi_s \in (a, b)$ .

### Přehled vzorců

Uvažujeme případ, kdy uzly  $x_i$  jsou ekvidistantní s krokem  $h$ , tj.  $x_i = x_0 + ih$ , kde  $i = 1, 2, \dots, n$ . Uzel, v němž počítáme přibližnou hodnotu derivace, si označíme jako  $x$  a ostatní s jeho pomocí jako  $x + h$ ,  $x - h$  apod.

Pomocí (1.6) dostaneme následující vztahy:

první dopředná diference:

$$f'(x) = \frac{f(x+h) - f(x)}{h} + \frac{1}{2}hf''(\xi), \quad \xi \in (x, x+h) \quad (1.7)$$

první zpětná diference:

$$f'(x) = \frac{f(x) - f(x-h)}{h} - \frac{1}{2}hf''(\xi), \quad \xi \in (x-h, x) \quad (1.8)$$

první centrální diference:

$$f'(x) = \frac{f(x+h) - f(x-h)}{2h} + \frac{1}{3}h^2f'''(\xi), \quad \xi \in (x-h, x+h) \quad (1.9)$$

*Poznámka.* Chyba ve zpětné a dopředné diferenci závisí na  $h$  a v centrální diferenci závisí na  $h^2$ . Vzhledem k těmto skutečnostem a k tomu, že  $h \rightarrow 0$ , je vhodnější pro numerické derivování používat centrální diferenci.

---

<sup>2</sup>= funkce  $f$  má spojitě derivace až do řádu  $n+1$  na intervalu  $\langle a, b \rangle$

Další užitečné vzorce odvozené z prvních diferencí:  
druhá centrální diference (lze odvodit užitím první dopředné a první zpětné diference):

$$f''(x) \approx \frac{f(x+h) - 2f(x) + f(x-h)}{h^2} \quad (1.10)$$

první centrální diference pro parciální derivace:

$$\frac{\partial f(x_1, x_2)}{\partial x_1} \approx \frac{f(x_1+h, x_2) - f(x_1-h, x_2)}{2h} \quad (1.11)$$

$$\frac{\partial f(x_1, x_2)}{\partial x_2} \approx \frac{f(x_1, x_2+h) - f(x_1, x_2-h)}{2h} \quad (1.12)$$

**První a druhé parciální derivace pro funkci n-proměnných**

$$\frac{\partial f(x_1, x_2, \dots, x_i, \dots, x_n)}{\partial x_i} \approx \frac{f(x_1, x_2, \dots, x_i+h, \dots, x_n) - f(x_1, x_2, \dots, x_i-h, \dots, x_n)}{2h}, \quad (1.13)$$

$$i = 1, 2, \dots, n$$

$$\frac{\partial^2 f(x_1, x_2, \dots, x_i, \dots, x_n)}{\partial x_i^2} \approx \frac{f(x_1, x_2, \dots, x_i+h, \dots, x_n) - 2f(x_1, x_2, \dots, x_i, \dots, x_n) + f(x_1, x_2, \dots, x_i-h, \dots, x_n)}{h^2}, \quad (1.14)$$

$$i = 1, 2, \dots, n$$

$$\begin{aligned} & \frac{\partial^2 f(x_1, \dots, x_i, \dots, x_j, \dots, x_n)}{\partial x_i \partial x_j} \approx \\ & \approx \frac{f(x_1, \dots, x_i+h, \dots, x_j, \dots, x_n) - f(x_1, \dots, x_i, \dots, x_n)}{h^2} + \\ & + \frac{-f(x_1, \dots, x_i+h, \dots, x_j-h, \dots, x_n) + f(x_1, \dots, x_i, \dots, x_j-h, \dots, x_n)}{h^2}, \end{aligned} \quad (1.15)$$

$$i, j = 1, 2, \dots, n$$

### 1.2.3 Minimalizace funkce jedné proměnné

Efektivní metody jednorozměrné minimalizace jsou důležité, protože jsou využívány v algoritmech vícerozměrné minimalizace, které opakovaně řeší úlohu určení délky kroku  $\lambda$  ve vztahu

$$\min_{\lambda > 0} (f(\mathbf{x}_k - \lambda \mathbf{g}(\mathbf{x}_k))) \quad (1.16)$$

#### Metoda kvadratické interpolace

Spočívá v aproximaci účelové funkce  $f(x)$  v okolí minima kvadratickou parabolou.

##### Algoritmus 1.2.1.

Na počátku výpočtu je třeba znát tři body  $0 \leq a_0 < c_0 < b_0$ , pro které platí  $f(a_0) \geq f(c_0)$  a  $f(c_0) \leq f(b_0)$ , zvolit si toleranci  $\varepsilon > 0$  a  $k := 0$ .

1. Body  $[a_k, f(a_k)]$ ,  $[c_k, f(c_k)]$ ,  $[b_k, f(b_k)]$  proložíme parabolou danou funkčním předpisem:

$$\begin{aligned}\varphi(x) &= Ax^2 + Bx + C \\ A &= -\frac{a_k f(c_k) - c_k f(a_k) - a_k f(b_k) + b_k f(a_k) + c_k f(b_k) - b_k f(c_k)}{(a_k - c_k)(a_k c_k - a_k b_k - c_k b_k + b_k^2)} \\ B &= \frac{a_k^2 f(c_k) - c_k^2 f(a_k) - a_k^2 f(b_k) + b_k^2 f(a_k) + c_k^2 f(b_k) - b_k^2 f(c_k)}{(a_k - c_k)(a_k c_k - a_k b_k - c_k b_k + b_k^2)} \\ C &= -\frac{a_k c_k^2 f(b_k) - a_k b_k^2 f(c_k) + c_k b_k^2 f(a_k) - a_k^2 c_k f(b_k) + a_k^2 b_k f(c_k) - c_k^2 b_k f(a_k)}{(a_k - c_k)(a_k c_k - a_k b_k - c_k b_k + b_k^2)}\end{aligned}\tag{1.17}$$

a určíme hodnotu  $x_k$ , pro které platí

$$f(x_k) = \min_x \varphi(x)$$

z podmínky  $\varphi'(x) = 0$ :

$$x_{k+1} := \frac{1}{2} \frac{a_k^2 f(c_k) - c_k^2 f(a_k) - a_k^2 f(b_k) + b_k^2 f(a_k) + c_k^2 f(b_k) - b_k^2 f(c_k)}{(a_k f(c_k) - c_k f(a_k) - a_k f(b_k) + b_k f(a_k) + c_k f(b_k) - b_k f(c_k))}\tag{1.18}$$

Dosažením dostaneme  $f(x_{k+1})$ .

2. Pokud  $x_{k+1} = c_k$ , potom zvolíme  $x_{k+1} := c_k - \frac{\delta}{2}$  nebo  $x_{k+1} := c_k + \frac{\delta}{2}$ , kde  $0 < \delta < (b_k - a_k)$ .
3. Pokud  $x_{k+1} < c_k$  a  $f(x_{k+1}) < f(c_k)$ , potom  $a_{k+1} := a_k$ ,  $c_{k+1} := x_{k+1}$ ,  $b_{k+1} := c_k$  a **GOTO 7.**
4. Pokud  $x_{k+1} < c_k$  a  $f(x_{k+1}) \geq f(c_k)$ , potom  $a_{k+1} := x_{k+1}$ ,  $c_{k+1} := c_k$ ,  $b_{k+1} := b_k$  a **GOTO 7.**
5. Pokud  $x_{k+1} > c_k$  a  $f(x_{k+1}) > f(c_k)$ , potom  $a_{k+1} := a_k$ ,  $c_{k+1} := c_k$ ,  $b_{k+1} := x_{k+1}$  a **GOTO 7.**
6. Pokud  $x_{k+1} > c_k$  a  $f(x_{k+1}) \leq f(c_k)$ , potom  $a_{k+1} := c_k$ ,  $c_{k+1} := x_{k+1}$ ,  $b_{k+1} := b_k$  a **GOTO 7.**
7. Pokud je splněna jedna z ukončovacích podmínek:

- (a)  $|x_{k+1} - x_k| < \varepsilon$   
 (b)  $|x_{k+1} - x_k| < \varepsilon|x_k|$   
 (c)  $|f(x_k) - f(x_{k+1})| < \varepsilon$   
 (d)  $|f(x_k) - f(x_{k+1})| < \varepsilon|f(x_k)|$

Potom **STOP.**

Jinak **GOTO 1.**



### 1.2.4 Minimalizace funkce více proměnných

#### Ukončovací podmínky

Všechny níže popsané metody jsou metody iterační, a proto musí mít nějaké ukončovací podmínky (kritéria):

1. Funkční hodnoty ve dvou po sobě následujících iteracích se málo liší:

$$|f(\mathbf{x}_k) - f(\mathbf{x}_{k+1})| < \varepsilon$$

pro dostatečně malé  $\varepsilon$

2. Body ve dvou po sobě následujících iterací jsou dostatečně blízko:

$$\|\mathbf{x}_k - \mathbf{x}_{k+1}\| < \varepsilon$$

pro dostatečně malé  $\varepsilon$ .

3. V metodách využívajících derivací účelové funkce: Gradient funkce je dostatečně malý:

$$\|\mathbf{g}_k\| < \varepsilon$$

pro dostatečně malé  $\varepsilon$

Další ukončovací podmínkou může být počet iterací, avšak tato podmínka není vhodná, protože pro každou funkci může metoda konvergovat jinou rychlostí. Kdybychom zvolili vysoký počet iterací, při kterých má dojít k ukončení metody, mohlo by to fungovat na velké množství metod, ale mohlo by docházet k hodně malým zlepšením a bylo by to tedy dost neefektivní.

#### Nelder-Mead

Tato metoda, také známá jako *metoda simplexů*, hledá minimum účelové funkce  $f$  porovnáváním jejích hodnot ve vrcholech simplexu<sup>3</sup> v prostoru  $\mathbb{R}^n$ . Pro  $n = 2$  se simplex nazývá trojúhelník, pro  $n = 3$  čtyřstěn.

**Algoritmus 1.2.2** (Nelder-Mead).

Na počátku výpočtu zvolíme počáteční aproximaci  $\mathbf{x}_0 = [x_1, x_2, \dots, x_n] \in \mathbb{R}^n$ , malé číslo  $\delta$  a požadované tolerance  $\varepsilon_1 > 0$  a  $\varepsilon_2 > 0$ . Další vrcholy 'startovacího' simplexu odvodíme z vrcholu  $\mathbf{x}_0$  takto:  $\mathbf{x}_i = [x_1, \dots, x_i + \delta, \dots, x_n]$ ,  $i = 1, 2, \dots, n$

1. Najdeme  $b$  takové, že

$$f(\mathbf{x}_b) = \max_{0 \leq j \leq n} f(\mathbf{x}_j)$$

Pokud  $\forall \mathbf{x}_i \neq \mathbf{x}_b$ ,  $i = 0, 1, \dots, n$  platí

(a)

$$\|\mathbf{x}_i - \mathbf{x}_b\| < \varepsilon_1$$

tj. vrcholy simplexu jsou navzájem dostatečně blízko, a zároveň

<sup>3</sup>simplex = n-rozměrné zobecnění trojúhelníku

(b)

$$|f(\mathbf{x}_i) - f(\mathbf{x}_b)| < \varepsilon_2$$

tj. funkční hodnoty vrcholů simplexu se málo liší,

potom **STOP** a  $\mathbf{x}_b$  prohlásíme za dostatečně dobrou aproximaci minima účelové funkce.

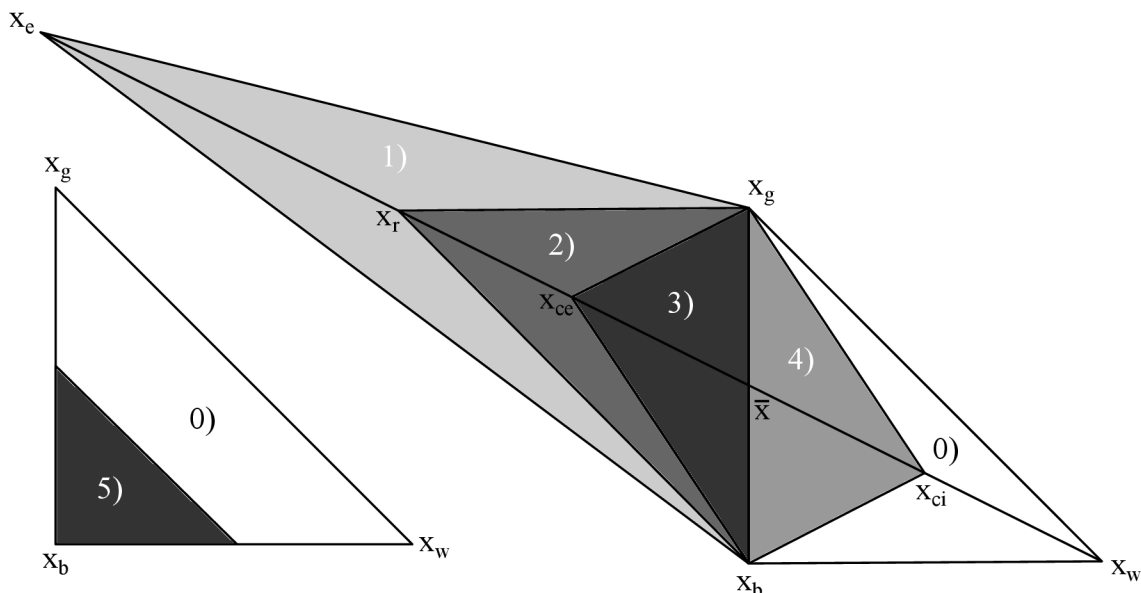
Jinak najdeme  $w$  takové, že

$$f(\mathbf{x}_w) = \min_{0 \leq j \leq n} f(\mathbf{x}_j)$$

a vypočteme těžiště:

$$\bar{\mathbf{x}} = \frac{1}{n} \sum_{j=0, j \neq w}^n \mathbf{x}_j$$

2. Označíme  $\mathbf{x}_r := \bar{\mathbf{x}} + (\bar{\mathbf{x}} - \mathbf{x}_w)$  jako reflexe a  $\mathbf{x}_e := \bar{\mathbf{x}} + 2(\bar{\mathbf{x}} - \mathbf{x}_w)$  jako expanze. Pokud  $f(\mathbf{x}_r) < f(\mathbf{x}_b) \wedge f(\mathbf{x}_e) < f(\mathbf{x}_b)$ , potom  $\mathbf{x}_w := \mathbf{x}_e$  a **GOTO 1.**
3. Pokud  $f(\mathbf{x}_r) < f(\mathbf{x}_g)$  pro nějaký vrchol  $\mathbf{x}_g \neq \mathbf{x}_w$ , potom  $\mathbf{x}_w := \mathbf{x}_r$  a **GOTO 1.**
4. Označíme  $\mathbf{x}_{ce} := \frac{1}{2}(\bar{\mathbf{x}} + \mathbf{x}_r)$  jako vnější kontrakce. Pokud  $f(\mathbf{x}_r) < f(\mathbf{x}_w) \wedge f(\mathbf{x}_{ce}) < f(\mathbf{x}_r)$ , potom  $\mathbf{x}_w := \mathbf{x}_{ce}$  a **GOTO 1.**
5. Označíme  $\mathbf{x}_{ci} := \frac{1}{2}(\bar{\mathbf{x}} + \mathbf{x}_w)$  jako vnitřní kontrakce. Pokud  $f(\mathbf{x}_r) \geq f(\mathbf{x}_w) \wedge f(\mathbf{x}_{ci}) < f(\mathbf{x}_w)$ , potom  $\mathbf{x}_w = \mathbf{x}_{ci}$  a **GOTO 1.**
6. Provedeme tzv. redukci:  $\mathbf{x}_i := \frac{1}{2}(\mathbf{x}_b + \mathbf{x}_i) \quad \forall \mathbf{x}_i \neq \mathbf{x}_b$  a **GOTO 1.**



Obr. 1.6: Nelder-Meadova metoda pro  $n = 2$ : 0) originální trojúhelník, 1) expanze, 2) reflexe, 3) vnější kontrakce, 4) vnitřní kontrakce, 5) redukce [3]

*Poznámka.* Tato metoda obsahuje velké množství výpočtů a v blízkosti minima je velmi pomalá a proto je vhodná pro  $n \leq 10$ .

### Metoda největšího spádu

Jedná se o základní minimalizační metodu, která využívá derivace účelové funkce ve formě gradientu. Patří tedy mezi tzv. gradientní metody.

Jak je známo funkce  $f(\mathbf{x})$  ve směru gradientu nejrychleji roste, tzn. nejrychleji klesá ve směru záporného gradientu. Gradient značíme:

$$\mathbf{g}(\mathbf{x}) \equiv \nabla f(\mathbf{x}) = \left( \frac{\partial f(\mathbf{x})}{\partial x_1}, \frac{\partial f(\mathbf{x})}{\partial x_2}, \dots, \frac{\partial f(\mathbf{x})}{\partial x_n} \right)^T$$

#### Algoritmus 1.2.3.

Na počátku výpočtu zvolíme počáteční aproximaci  $\mathbf{x}_0 = [x_1, x_2, \dots, x_n] \in \mathbb{R}^n$ , požadovanou toleranci  $\varepsilon > 0$  a  $k := 0$

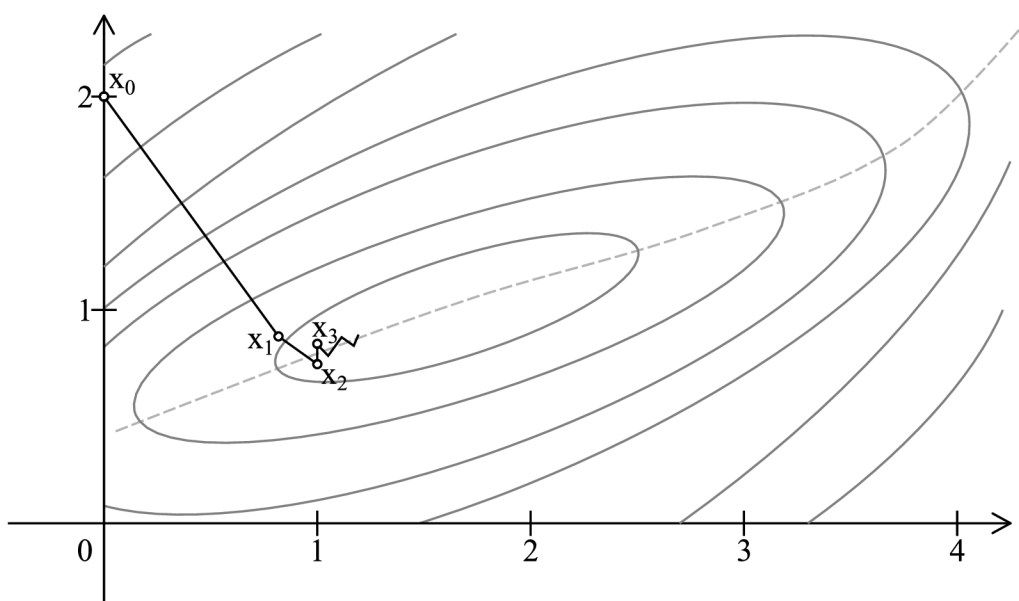
1. Pokud je  $\|\mathbf{g}(\mathbf{x}_k)\| < \varepsilon$ , potom **STOP**.

Jinak přiřadíme  $\mathbf{x}_{k+1} := \mathbf{x}_k - \lambda_k \mathbf{g}(\mathbf{x}_k)$ , kde  $\lambda_k \approx \min_{\lambda > 0} (f(\mathbf{y}_k + \lambda \mathbf{e}_k))$ ,  $k := k + 1$  a **GOTO 1.**

*Poznámka.* Pokud využíváme normovaného gradientu, pak nám parametr délky kroku  $\lambda_k$  určuje vzdálenost bodů  $\mathbf{x}_k$  a  $\mathbf{x}_{k+1}$ .

Pro hledání délky kroku, které vede na jednorozměrnou minimalizaci, je dostačující využití pouze prvního kroku kvadratické interpolace.

*Poznámka.* Metoda největšího spádu konverguje k řešení poměrně rychle, ale v blízkosti minima je velmi pomalá.



Obr. 1.7: Ilustrace metody největšího spádu [7]

### Newtonova metoda

Tato metoda je gradientní metodou, která hledá minimum jak řešení soustavy nelineárních rovnic  $\mathbf{g}(\mathbf{x}) = \mathbf{o}$  Newtonovou metodou. Takovým řešením může být stacionární bod (maximum, minimum, ale také sedlový bod).

K řešení budeme potřebovat tzv. *Hessovu matici* funkce  $f$ :

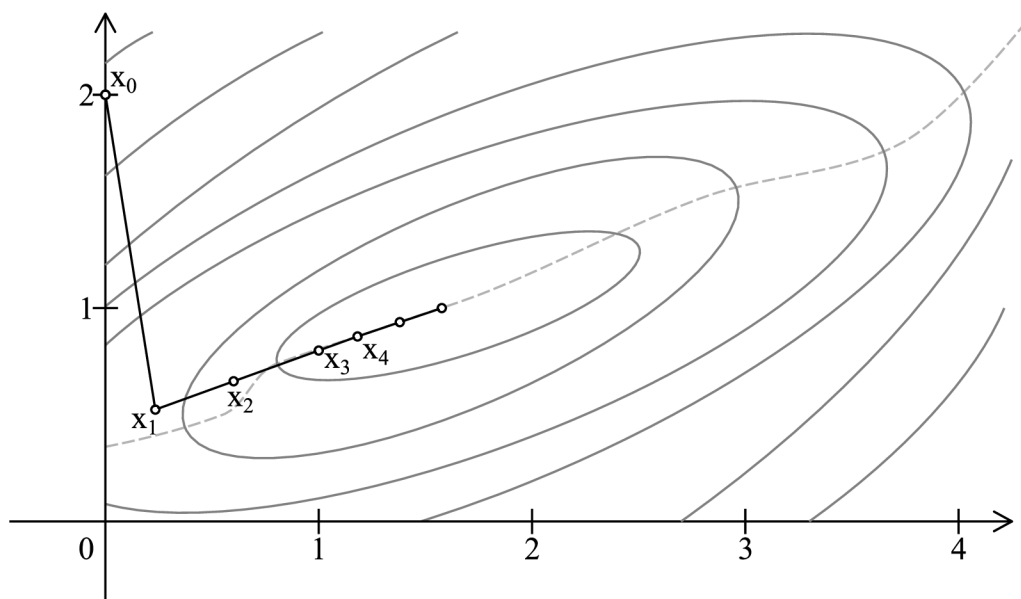
$$\mathbf{g}'(\mathbf{x}) \equiv \mathbf{H}(\mathbf{x}) = \begin{pmatrix} \frac{\partial^2 f(\mathbf{x})}{\partial x_1^2} & \frac{\partial^2 f(\mathbf{x})}{\partial x_1 \partial x_2} & \cdots & \frac{\partial^2 f(\mathbf{x})}{\partial x_1 \partial x_n} \\ \frac{\partial^2 f(\mathbf{x})}{\partial x_2 \partial x_1} & \frac{\partial^2 f(\mathbf{x})}{\partial x_2^2} & \cdots & \frac{\partial^2 f(\mathbf{x})}{\partial x_2 \partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial^2 f(\mathbf{x})}{\partial x_n \partial x_1} & \frac{\partial^2 f(\mathbf{x})}{\partial x_n \partial x_2} & \cdots & \frac{\partial^2 f(\mathbf{x})}{\partial x_n^2} \end{pmatrix}$$

#### Algoritmus 1.2.4.

Na počátku výpočtu zvolíme počáteční aproximaci  $\mathbf{x}_0 = [x_1, x_2, \dots, x_n] \in \mathbb{R}^n$ , požadovanou toleranci  $\varepsilon > 0$  a  $k := 0$

1. Pokud je  $\|\mathbf{g}(\mathbf{x}_k)\| < \varepsilon$ , potom **STOP**.

Jinak přiřadíme  $\mathbf{x}_{k+1} := \mathbf{x}_k + \mathbf{d}_k$ , kde  $\mathbf{d}_k$  je řešením lineární soustavy rovnic  $\mathbf{H}(\mathbf{x}_k)\mathbf{d}_k = -\mathbf{g}(\mathbf{x}_k)$  a  $k := k + 1$  a **GOTO 1.**



Obr. 1.8: Ilustrace Newtonovy metoda [7]

*Poznámka.* Směrový vektor  $\mathbf{d}_k$  není obecně spádový, ale v blízkosti minima spádový je.

*Poznámka.* Pokud je počáteční aproximace dostatečně blízko řešení konverguje tato metoda k řešení velmi rychle (kvadraticky), v případě, že není blízko řešení, nemusí však konvergovat vůbec. Vzhledem k těmto skutečnostem a k tomu, že metoda největšího spádu konverguje daleko od minima velmi rychle a v blízkosti minima konverguje velmi pomalu, je vhodné začít výpočet metodou největšího spádu a dokončit ho Newtonovou metodou.

## 1.3 Konvoluce funkce dvou proměnných

**Definice 1.21.** Necht  $f(x, y)$  a  $g(x, y)$  jsou reálné funkce reálných proměnných. Funkci  $h(x, y)$ , pro kterou platí:

$$h(x, y) = f(x, y) * g(x, y) = \iint_{\mathbb{R} \times \mathbb{R}} f(u, v)g(x - u, y - v) du dv \quad (1.19)$$

nazýváme *konvoluce funkce  $f(x, y)$  s konvolučním jádrem  $g(x, y)$* .

Vlastnosti konvoluce:

(K1)  $\forall f(x, y), g(x, y) \in \mathbb{R}^2 :$

$$f * g = g * f$$

(komutativita)

(K2)  $\forall f(x, y), g(x, y) \in \mathbb{R}^2, \forall a, b \in \mathbb{R} :$

$$af * bg = ab(f * g)$$

(násobení konstantou)

(K3)  $\forall f(x, y), g_1(x, y), g_2(x, y) \in \mathbb{R}^2 :$

$$f * (g_1 + g_2) = f * g_1 + f * g_2$$

(distributivita vůči sčítání)

(K4)  $\forall f_1(x, y), f_2(x, y), f_3(x, y) \in \mathbb{R}^2 :$

$$f_1 * (f_2 * f_3) = (f_1 * f_2) * f_3$$

(asociativita)

### 1.3.1 Diskrétní konvoluce

**Definice 1.22.** Necht  $f(x, y)$  a  $g(x, y)$  jsou reálné funkce, kde proměnné  $x, y \in \mathbb{Z}$ . Funkci  $h(x, y)$ , pro kterou platí:

$$h(x, y) = f(x, y) * g(x, y) = \sum_k \sum_l f(k, l)g(x - k, y - l) \quad (1.20)$$

nazýváme *diskrétní konvoluce*.

*Poznámka.* V programátorské praxi je vhodnější využít komutativity konvoluce a místo (1.20) používat diskrétní konvoluci ve tvaru

$$h(x, y) = f(x, y) * g(x, y) = \sum_k \sum_l f(x - k, y - l)g(k, l) \quad (1.21)$$

### 1.3.2 Neúplná konvoluce

V některých případech může dojít k tomu, že budeme znát funkční hodnoty jen v některých částech definičního oboru funkce  $f(x, y)$ . Pro konvoluci tedy bereme pouze ty části, které jsou definovány.

**Definice 1.23.** Nechť  $f(x, y)$  a  $g(x, y)$  jsou reálné funkce reálných proměnných. Nechť  $D(f) \subset \mathbb{R}^2$  a  $D(g) = \mathbb{R}^2$ . Funkci  $h(x, y)$ , pro kterou platí:

$$h(x, y) = f(x, y) * g(x, y) = \frac{\iint_{D(f)} f(u, v)g(x - u, y - v) du dv}{\iint_{D(f)} g(x - u, y - v) du dv} \quad (1.22)$$

nazýváme *neúplná konvoluce funkce  $f(x, y)$  s konvolučním jádrem  $g(x, y)$* .

**Definice 1.24.** Nechť  $f(x, y)$  a  $g(x, y)$  jsou reálné funkce, kde proměnné  $x, y \in \mathbb{Z}$ . Nechť  $D(f) \subset \mathbb{Z}^2$  a  $D(g) = \mathbb{Z}^2$ . Funkci  $h(x, y)$ , pro kterou platí:

$$hS(x, y) = f(x, y) * g(x, y) = \frac{\sum \sum_{k, l \in D(f)} f(x - k, y - l)g(k, l)}{\sum \sum_{k, l \in D(f)} g(k, l)} \quad (1.23)$$

nazýváme *diskrétní neúplná konvoluce funkce  $f(x, y)$  s konvolučním jádrem  $g(x, y)$* .

## 2 ZPRACOVÁNÍ OBRAZOVÉ INFORMACE

Tato kapitola se věnuje definicím pojmů pro zpracování obrazu, jejich vlastností a metodám jeho zpracování. K tomu jsme čerpali z [8], [12], [13], [14] a [2].

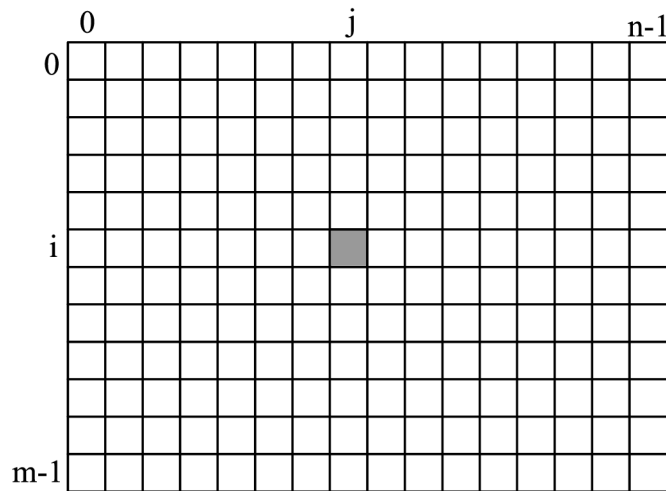
### 2.1 Obraz a jeho vlastnosti

#### 2.1.1 Obrazová matice

Obraz matematicky chápeme jako spojitou funkci  $f(x, y)$ ,  $x, y \in \mathbb{R}$ . Funkční hodnota  $f(x, y)$  představuje hodnotu např. jasu, barvy, teploty, záření, intenzity světla atd.

V dnešní době se však setkáváme především s digitálním obrazem. Digitální obraz představuje matice o rozměrech  $m \times n$ , kterou nazýváme *obrazová matice* a značíme  $\mathcal{I}_{m \times n}$ .

Jeden prvek obrazové matice nazýváme *obrazový element* nebo *pixel*<sup>1</sup>. Pixely mají obvykle tvar čtverce nebo obdélníku a jejich polohu určuje řádkový index  $i$  a sloupcový index  $j$  (viz Obrázek 2.1). Pixel  $\mathcal{P}$  obrazové matice  $\mathcal{I}_{m \times n}$  o souřadnicích  $[i, j]$ , kde  $0 \leq i \leq m - 1$  a  $0 \leq j \leq n - 1$  značíme  $\mathcal{P}[i, j]$ .



Obr. 2.1: Obrazová matice [8]

**Definice 2.1.** Necht  $\mathcal{I}_{m \times n} = (\mathcal{P}[i, j])$  je obrazová matice a  $r > 0$ . Pak množinu

$$\mathcal{O} = \{\mathcal{P}[k, l] : \sqrt{(k - i)^2 + (l - j)^2} \leq r\}$$

nazýváme *kruhové okolí pixelu*  $\mathcal{P}[i, j]$  a píšeme  $\mathcal{O}_{\circ}^{\mathcal{P}}(i, j, r)$ .

**Definice 2.2.** Necht  $\mathcal{I}_{m \times n} = (\mathcal{P}[i, j])$  je obrazová matice a  $r > 0$ . Pak množinu

$$\mathcal{O} = \{\mathcal{P}[k, l] : k, l \in \langle i - r, i + r \rangle\}$$

nazýváme *čtvercové okolí pixelu*  $\mathcal{P}[i, j]$  a píšeme  $\mathcal{O}_{\square}^{\mathcal{P}}(i, j, r)$ .

<sup>1</sup>pixel = picture element

## 2.1.2 Atributy pixelu

Pixel  $\mathcal{P}$  může obecně nabývat nějaké  $n$ -tice hodnot. Nejčastěji u barevného obrazu nabývá trojice hodnot: intenzity červené, zelené a modré složky světla. Z těchto hodnot lze určit další vlastnosti pixelu, které nazýváme *atributy*.

Atributy pixelu dělíme na:

- *vlastní atributy* - lze je určit ze samotného pixelu
- *nevlastní atributy* - určují se dle nějakého daného okolí

### Vlastní atributy

Základní vlastní atributy:

*Intenzita složek červené  $R$ , zelené  $G$  a modré  $B$  barvy,*

kde  $R, G, B \in \langle 0, D \rangle$ , kde  $D = 2^n - 1$ , kde  $n$  je barevná hloubka<sup>2</sup>. Pro 8-bitový obraz je  $D = 255$ , pro 16-bitový  $D = 65535$ .

*Jas pixelu  $J$ :*

$$J = \frac{R + G + B}{3}, \quad J \in \langle 0, D \rangle \quad (2.1)$$

*Normalizovaný jas pixelu  $J_n$ :*

$$J_n = \frac{J}{D}, \quad J_n \in \langle 0, 1 \rangle \quad (2.2)$$

*Tón barvy  $H$  (tzv. úhel barvy udáván ve stupních  $[\circ]$ ):*

$$H = \begin{cases} \max = \max\{R, G, B\} \\ \min = \min\{R, G, B\} \\ \left. \begin{array}{l} 60^\circ \cdot \left(\frac{G-B}{\max-\min}\right), \quad \max = R \wedge G \geq B \\ 60^\circ \cdot \left(\frac{G-B}{\max-\min} + 6\right), \quad \max = R \wedge B > G \\ 60^\circ \cdot \left(\frac{B-R}{\max-\min} + 2\right), \quad \max = G \\ 60^\circ \cdot \left(\frac{R-G}{\max-\min} + 4\right), \quad \max = B \\ \text{nedefinován,} \quad \max = \min \end{array} \right\} \quad (2.3) \\ H \in \langle 0, 360 \rangle \end{cases}$$

*Saturace barvy  $S$ :*

$$S = 1 - \frac{\min\{R, G, B\}}{\max\{R, G, B\}}, \quad \max\{R, G, B\} \neq 0, \quad S \in \langle 0, 1 \rangle \quad (2.4)$$

*Světlost pixelu  $L$ :*

$$L = \frac{1}{2}(\max\{R, G, B\} + \min\{R, G, B\}), \quad L \in \langle 0, D \rangle \quad (2.5)$$

*Poznámka.* V případě obrazu v odstínech šedé, tzv. *černobílém* obrazu platí:  $R = G = B = J = L$ ,  $H = \text{nedef.}$  a  $S = 0$ , je tedy reprezentován pouze jednou jasovou hodnotou.

<sup>2</sup>= počet bitů použitých k popisu jednoho pixelu



### Nevlastní atributy

Tyto atributy jsou závislé na velikosti a tvaru uvažovaného okolí.

Základní nevlastní atributy pixelu  $\mathcal{P}[i, j]$ :

*Střední hodnota*  $E(\mathcal{P}[i, j])$ :

$$E(\mathcal{P}[i, j]) = \frac{1}{n} \sum_{k=1}^n \mathcal{P}_k, \quad (2.6)$$

kde  $\mathcal{P}_k \in \mathcal{O}^{\mathcal{P}}(i, j, r)$ , kde  $\cdot \in \{\square, \circ\}$ , a  $\text{card } \mathcal{O}^{\mathcal{P}}(i, j, r) = n$ .

*Rozptyl hodnot*  $D(\mathcal{P}[i, j])$ :

$$D(\mathcal{P}[i, j]) = \frac{1}{n} \sum_{k=1}^n (\mathcal{P}_k - E(\mathcal{P}[i, j]))^2, \quad (2.7)$$

kde  $\mathcal{P}_k \in \mathcal{O}^{\mathcal{P}}(i, j, r)$ , kde  $\cdot \in \{\square, \circ\}$ , a  $\text{card } \mathcal{O}^{\mathcal{P}}(i, j, r) = n$ .

## 2.2 Barevné modely

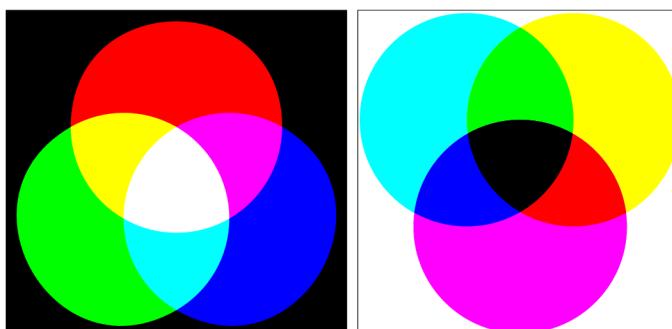
Barevný model určuje jakým způsobem se tvoří výsledná barva každého pixelu. Používá základní barvy a mísením těchto barev vznikají další barvy.

### 2.2.1 Aditivní model

V těchto modelech se jednotlivé základní barvy sčítají a výsledkem je světlo větší intenzity. Základními barvami tohoto modelu jsou červená  $R$ , zelená  $G$  a modrá  $B$ . Barvy vznikají dle obrázku 2.2(a). Tento model využívají veškerá aktivní zobrazovací zařízení jako např. televize a monitory.

### 2.2.2 Subtraktivní model

V těchto modelech se s každou přidanou barvou ubírá část původního světla - světlo prochází jednotlivými barevnými vrstvami a je více pohlcováno. Základními barvami tohoto modelu jsou azurová  $C$ , purpurová  $M$  a žlutá  $Y$ . Barvy vznikají dle obrázku 2.2(b). Tento model využívají zejména barevné tiskárny, kdy tisk probíhá na bílý papír.



(a) Aditivní RGB

(b) Subtraktivní CMY

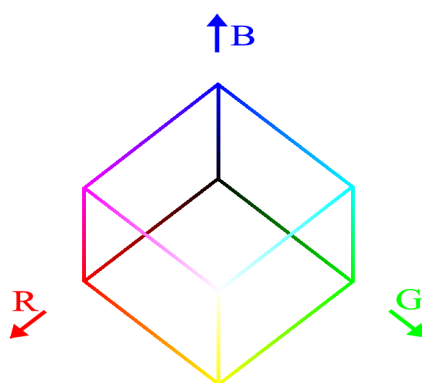
Obr. 2.2: Barevné modely [13]

## 2.3 Barevné prostory

Pro práci s barvou je však vhodné si zavést tzv. *barevný prostor*. Barevný prostor je předem definovaná množina barev, kterou je schopno zařízení snímat, zobrazit nebo reprodukovat.

### 2.3.1 RGB prostor

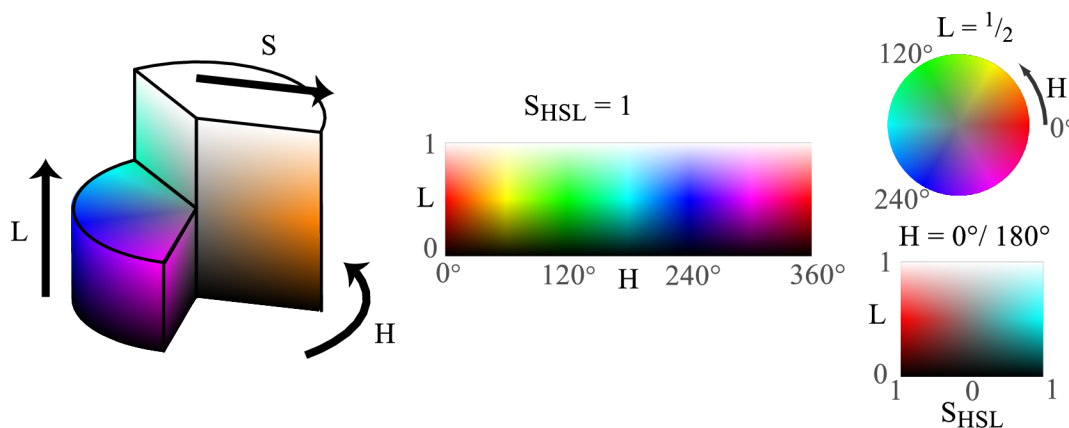
Základním barevným prostorem je RGB prostor, kdy se intenzity jednotlivých složek nanášejí v kartézském souřadném systému. Vzhledem k tomu, že intenzita R, G, B může nabývat hodnot  $\langle 0, D \rangle$ , kde  $D = 2^n - 1$ , kde  $n$  je barevná hloubka. Tímto prostorem je tedy krychle (viz obrázek 2.3).



Obr. 2.3: RGB krychle [14]

### 2.3.2 HSL prostor

Vzhledem k tomu, že z R, G, B lze odvodit další atributy pixelu, je možné definovat tzv. HSL prostor ( $H$  = tón barvy,  $S$  = saturace,  $L$  = světlost pixelu). Protože  $H$  je definován jako úhel barvy, lze tento prostor zakreslit do cylindrického souřadného systému a HSL prostorem je tedy válec (viz obrázek 2.4).



Obr. 2.4: HSL prostor [14]

## 2.4 Kalibrace digitálních obrazů

Reálný obraz je zatížen různými vadami způsobenými nehomogenním osvětlením pozorovaného vzorku, vinětačí optické soustavy, impulsním šumem atd.. Tyto vady lze pomocí kalibrace potlačit a tím zvýšit kvalitu obrazu pro jeho další zpracování.

### 2.4.1 Základní typy obrazů

Při popisu obrazů se používají dva parametry:

*Teplota*  $T$  - při které je pořízen.

*Čas*  $t$  - tzv. expoziční, tj. doba, po kterou byl obraz snímán.

#### Light-Image

- obraz, který chceme kalibrovat. Značíme ho  $L(T, t)$ .

#### Dark-frame

- obraz pořízený v absolutní tmě, značíme ho  $D(T, t)$ .

#### Bias-frame

- dark-frame s nulový expozičním časem, tedy  $D(T, 0)$ . Značíme ho  $B(T)$ . V praxi není možné dosáhnout nulového expozičního času a proto za něj považujeme nejmenší dosažený expoziční čas.

#### Flat-field

- obraz dokonale homogenní rovnoměrně osvětlené šedé plochy, který není saturovaný, tj. ani jeden pixel není saturován. Značíme ho  $F(T, t)$ .

### 2.4.2 Metody kalibrace

#### *Odečtení dark-frame*

Dark-frame je třeba vytvořit ihned po vytvoření light-image, aby byla dodržena stejná teplota.

- Nejnižší nezbytně nutná metoda kalibrace je:

$$A = L(T, t) - B(T)$$

- Lepší kalibrace nutná pro delší expoziční časy je:

$$A = L(T, t) - D(T, t)$$

*Korekce flat-fieldem*

Tato metoda koriguje nerovnoměrné osvětlené obrazové roviny, různou citlivost fotodiód na světlo a odstraní tzv. vinětaci - úbytek světla se vzrůstající vzdáleností od optické osy.

$$A = \frac{L(T_1, t_1) - D(T_1, t_1)}{F(T_2, t_2) - D_F(T_2, t_2)} \cdot k,$$

kde  $D_F(T, t)$  je dark-frame pro flat-field, tzv. *dark-flat* a  $k$  slouží k nastavení hodnot ve výsledném rozsahu tak, aby byl využit celý dynamický rozsah.

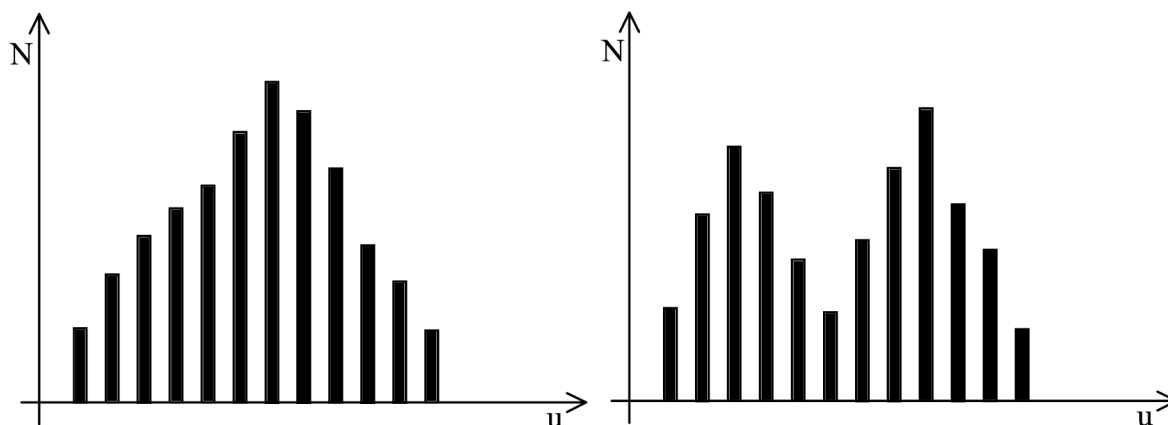
## 2.5 Histogram

V obrazové analýze se často využívají tzv. histogramy, které jsou definovány následovně:

**Definice 2.3.** Jako *histogram* nazýváme funkci  $N$  četnosti výskytu určité úrovně  $u$ :

$$N = N(u)$$

Histogram ukazuje, jak je využit dynamický rozsah a zda všechny úrovně jsou obsazeny se stejnou četností.



Obr. 2.5: Typické případy histogramu [8]

## 2.6 Lineární filtry

*Poznámka.* Pro snazší popis je vhodné uvažovat, že každý pixel je reprezentován pouze jednou hodnotou. To lze, protože barevný obraz můžeme reprezentovat třemi obrazovými maticemi (pro každou barvu zvlášť):  $\mathcal{I}_{m \times n}^R$ ,  $\mathcal{I}_{m \times n}^G$ ,  $\mathcal{I}_{m \times n}^B$ .

Lineární filtry mají v obrazové analýze široké uplatnění. Jejich vlastnosti se mohou velmi lišit a to pouze zvolením různých konvolučních jader. Lineární filtry upravují frekvenční spektrum zpracovaného obrazu.

**Definice 2.4.** Necht  $\mathcal{I}_{h \times w}^{in} = (\mathcal{P}[i, j])$  a  $\mathcal{I}_{h \times w}^{out} = (\mathcal{Q}[i, j])$  jsou obrazové matice. Necht  $\mathbf{G} = (g[k, l])$ , kde  $0 \leq k \leq 2m$  a  $0 \leq l \leq 2n$ . Zobrazení  $\mathcal{F}_{lin} : \mathcal{P}[i, j] \rightarrow \mathcal{Q}[i, j]$ , pro které platí:

$$\mathcal{Q}[i, j] = \sum_{k=-m}^m \sum_{l=-n}^n \mathcal{P}[i - k, j - l] g[k + m, l + n] \quad (2.8)$$

nazýváme *lineární filtr* s konvolučním jádrem  $\mathbf{G}$ .

### 2.6.1 Základní typy lineárních filtrů

#### Filtr typu identický obraz

Tento filtr nepotlačuje žádné frekvence obrazu. Platí  $\mathcal{I}_{h \times w}^{in} = \mathcal{I}_{h \times w}^{out}$ .

Konvoluční jádro tohoto filtru je:

$$\mathbf{G}_I = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad (2.9)$$

#### Filtr typu dolní propust

Tento filtr by měl propouštět nízké frekvence a tedy odstraňovat vyšší prostorové frekvence. Vyšší prostorové frekvence má např. šum.

Nejjednodušší konvoluční jádro tohoto filtru tvoří tzv. klouzavý průměr:

$$\mathbf{G}_M = \frac{1}{9} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} \quad (2.10)$$

Filtr s tímto konvolučním jádrem propouští velmi nízké frekvence, ale mírně i vyšší frekvence, což však je u tohoto filtru nežádoucí. Je proto vhodnější využít konvoluční jádro, jehož prvky jsou určeny gaussovou funkcí dvou proměnných.

*Poznámka.* Gaussova funkce dvou proměnných má tvar:

$$f(x, y) = a \exp \left\{ - \left( \frac{(x^2 - \mu_x)}{2\sigma_x^2} + \frac{(y^2 - \mu_y)}{2\sigma_y^2} \right) \right\},$$

kde  $a, \sigma_x, \sigma_y > 0$  a  $\mu_x, \mu_y \in \mathbb{R}$ .

Graf této funkce má v bodě  $[\mu_x, \mu_y]$  vrchol s funkční hodnotou  $a$ .  $\sigma_x, \sigma_y$  určují „šířku“ na osách  $x$  a  $y$ , kde funkční hodnoty jsou větší než „přibližná“ nula (tj. přibližně větší než 0.009).

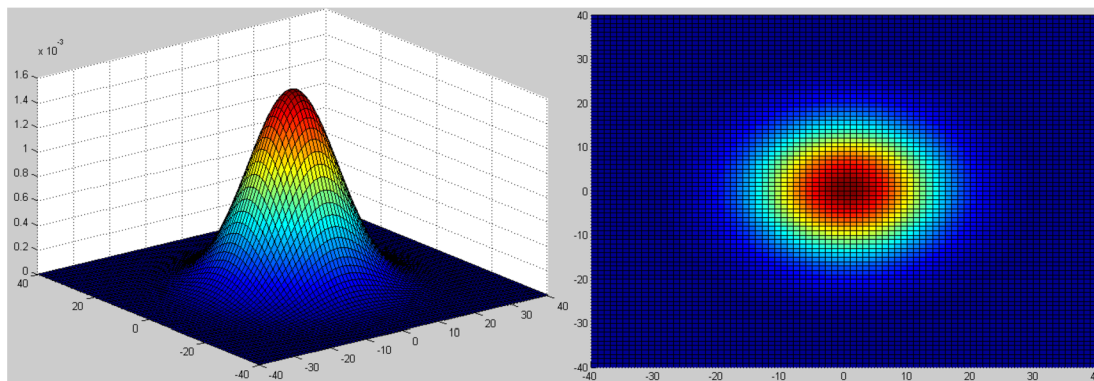
Pro naše využití je vhodné volit  $\mu_x = \mu_y = 0$ ,  $\sigma_x = \sigma_y = \sigma$  a  $a = 1$  případně tak, aby součet všech prvků jádra byl roven jedné (především pro filtry typu dolní propust a horní propust), tedy

$$a = \frac{1}{\sum_{k=-m}^m \sum_{l=-n}^n g[k + m, l + n]}.$$

Gaussova funkce pro naše použití má tvar:

$$f(x, y) = a \exp \left\{ -\frac{x^2 + y^2}{2\sigma^2} \right\}, \quad (2.11)$$

kde  $a, \sigma > 0$ .



Obr. 2.6: Konvoluční gaussovské jádro typu dolní propust o velikosti  $81 \times 81$  se  $\sigma = 10$

Konvoluční jádro typu dolní propust o velikosti  $5 \times 5$  určené gaussovou funkcí dvou proměnných, kde  $\sigma = 1$  vypadá následovně:

$$\mathbf{G}_L = \frac{1}{6,1688} \begin{bmatrix} 0,0183 & 0,0821 & 0,1353 & 0,0821 & 0,0183 \\ 0,0821 & 0,3679 & 0,6065 & 0,3679 & 0,0821 \\ 0,1353 & 0,6065 & 1 & 0,6065 & 0,1353 \\ 0,0821 & 0,3679 & 0,6065 & 0,3679 & 0,0821 \\ 0,0183 & 0,0821 & 0,1353 & 0,0821 & 0,0183 \end{bmatrix} \quad (2.12)$$

### Filtr typu horní propust

Tento filtr propouští vysoké frekvence a tedy odstraňuje nižší prostorové frekvence.

Konvoluční jádro filtru typu horní propust lze určit z předchozích typů takto:

$$\mathbf{G}_H = \mathbf{G}_I - \mathbf{G}_L$$

Po dosazení (2.12):

$$\mathbf{G}_H = \frac{1}{6,1688} \begin{bmatrix} -0,0183 & -0,0821 & -0,1353 & -0,0821 & -0,0183 \\ -0,0821 & -0,3679 & -0,6065 & -0,3679 & -0,0821 \\ -0,1353 & -0,6065 & 5,1688 & -0,6065 & -0,1353 \\ -0,0821 & -0,3679 & -0,6065 & -0,3679 & -0,0821 \\ -0,0183 & -0,0821 & -0,1353 & -0,0821 & -0,0183 \end{bmatrix} \quad (2.13)$$

*Poznámka.* V praktických úlohách se používají nejen čtvercová konvoluční jádra, ale i obdélníková, kruhová, eliptická aj. Zvolený tvar závisí přímo na řešené úloze.

*Příklad* (kruhového jádra). Necht  $r > 0$  je velikost kruhového konvolučního jádra. Pak pro prvky konvoluční matice  $\mathbf{G}$  platí:  $\forall i \in \langle 0, 2m \rangle$  a  $\forall j \in \langle 0, 2n \rangle$  splňující  $\sqrt{(m-i)^2 + (n-j)^2} \leq r$  platí  $g[i, j] = 0$ .

Aproximace kruhového jádra se stejnými hodnotami jako (2.12):

$$\mathbf{G}_L = \frac{1}{5,4388} \begin{bmatrix} 0 & 0 & 0,1353 & 0 & 0 \\ 0 & 0,3679 & 0,6065 & 0,3679 & 0 \\ 0,1353 & 0,6065 & 1 & 0,6065 & 0,1353 \\ 0 & 0,3679 & 0,6065 & 0,3679 & 0 \\ 0 & 0 & 0,1353 & 0 & 0 \end{bmatrix} \quad (2.14)$$





### 3 REKONSTRUKCE POVRCHU OBJEKTU

Metoda je založena na záměrné barevné vadě objektivu konfokálního mikroskopu. Rozdílná vzdálenost povrchu od objektivu se projevuje rozdílnou barvou odraženého světla od povrchu objektu. V rámci jednoho snímku tedy lze na základě barvy určit jeho výškový profil. K tomuto je nutné znát závislost mezi barvou a výškou. Lze předpokládat, že tato závislost nebude lineární.

Námi vytvořené numerické metody zpracování obrazové informace využívají 16-bitové snímky.

#### 3.1 Závislost výšky povrchu na barvě

K zjištění závislosti výšky na barvě je nutné pořídit vhodné kalibrační snímky. Tyto snímky lze použít pouze pro naši optickou soustavu, jejíž vlastnosti ovlivňuje typ objektivu, typ kamery a zdroj osvětlení. V případě jiné optické soustavy musíme vytvořit jiné kalibrační snímky i snímek pro rekonstrukci povrchu.

Jako kalibrační snímky jsme použili snímky odrazu zrcátka, u kterého lze předpokládat lokálně konstantní výšku povrchu (viz obrázky 3.1). Tyto snímky mají stejnou velikost:  $1304 \times 976$  px a jsou kalibrovány odečtením dark-framu. Použití korekce flat-fieldem není v tomto případě nutné, jelikož nás zajímá pouze barva pixelu. Posunutí mezi jednotlivými snímky je  $0,005$  mm.

Z těchto snímků vybereme jeden, ze kterého určíme souřadnice pixelu a v každém kalibračním snímku vybereme pixel s těmito souřadnicemi s jeho okolím. Z tohoto vzorku zjistíme hodnoty tónu barvy  $H$  a ty nanese do grafu závislosti  $H(x)$  tónu barvy  $H$  na pořadí snímku (= relativní výšky)  $x$ . Pro zjištění funkční závislosti využijeme průměr těchto hodnot a na vzniklé body aplikujeme MNČ (viz kapitola 1.2.1).

Vlivem nedokonalosti objektivu je v každé části obrazu mírně jiná závislost barvy na vzdálenosti povrchu od objektivu (viz obrázek 3.2), což má dále za následek chybné určení výšky povrchu. Pro jednoduchost řešení úlohy budeme předpokládat, že tyto rozdíly jsou minimální a tuto vadu zanedbáme. Pro kalibraci však vybereme z referenčního snímku pixely se stejným tónem barvy.

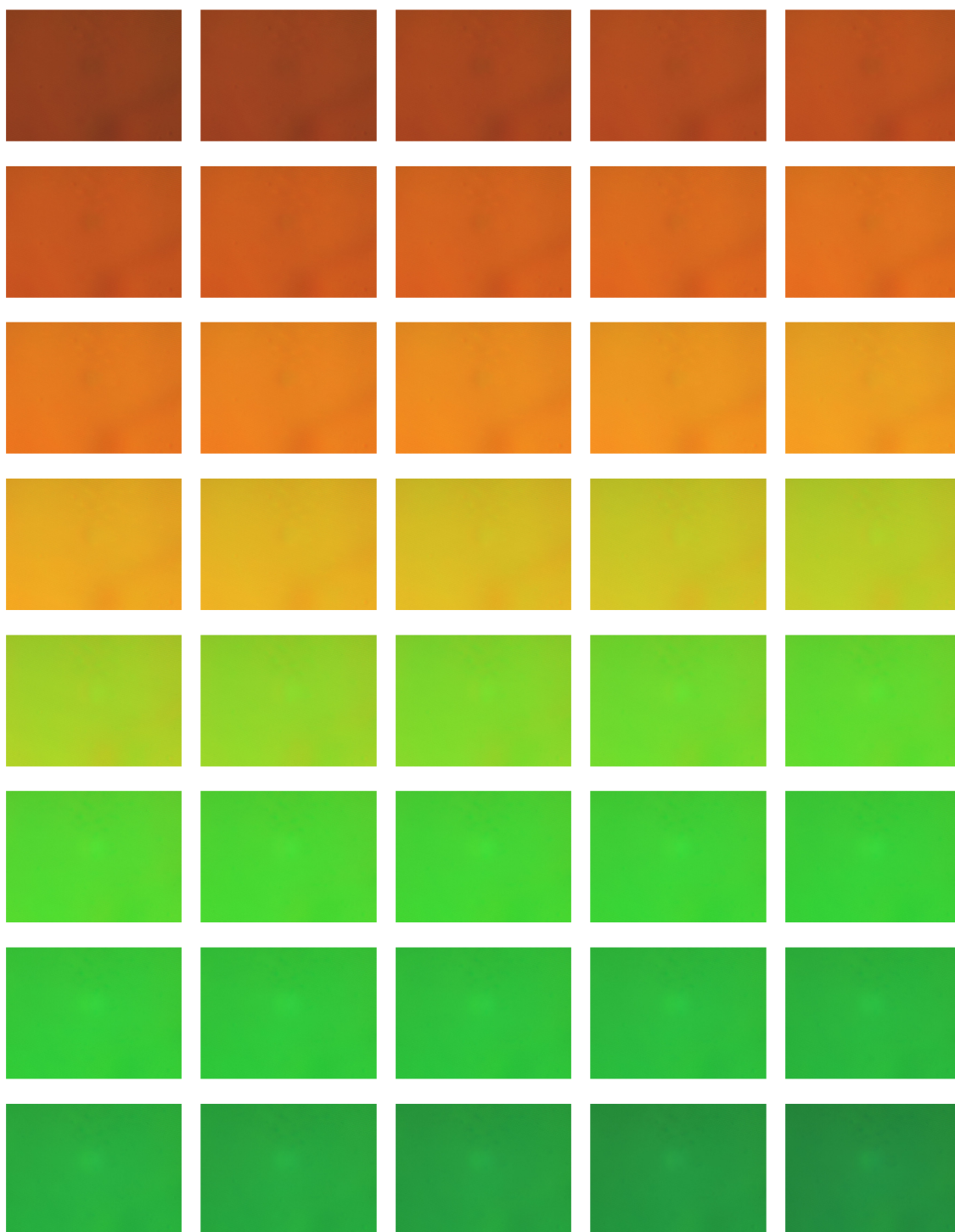
Na obrázku 3.3 je znázorněna bodová závislost výšky na barvě. Tato závislost velmi připomíná funkci  $\arctan x$ . Pro aproximaci naměřených hodnot jsme použili následující funkci:

$$H(x) = C \left( \frac{\arctan(Ax + B)}{\pi} + \frac{1}{2} \right)^E + D \quad (3.1)$$

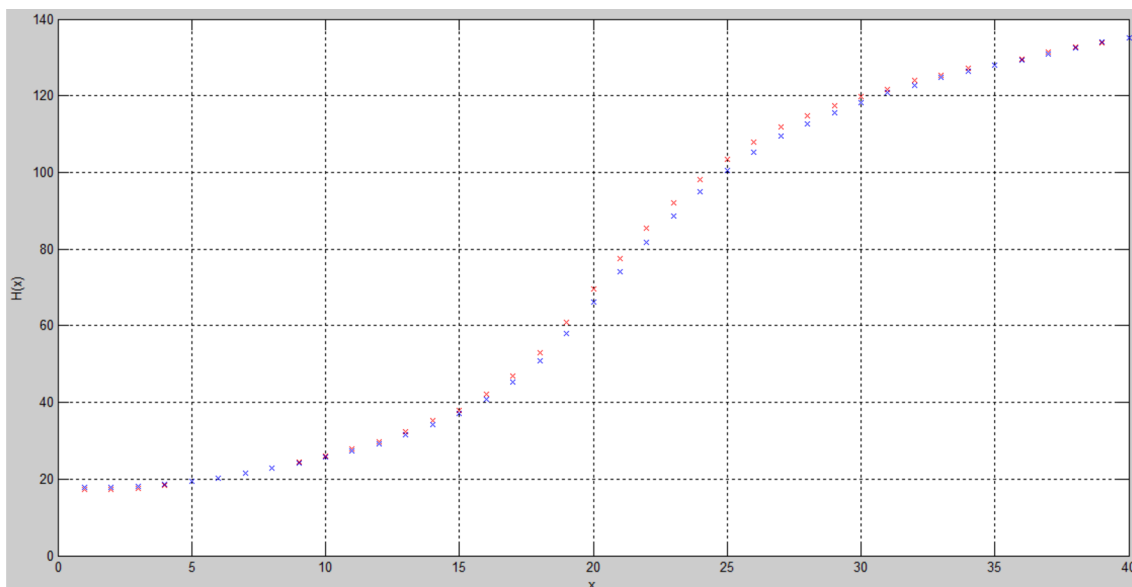
Pro nalezení optimálních hodnot koeficientu  $A$ ,  $B$ ,  $C$ ,  $D$ ,  $E$  jsme použili metodu nejmenších čtverců. Vzhledem k tomu, že funkce není z hlediska koeficientů lineární, museli jsme použít pro hledání minima součtu kvadrátů reziduí metodu vícerozměrné optimalizace. Prakticky jsme realizovali metodu Nelder-Mead, která však nevedla k zjištění

globálního minima, protože často konvergovala k lokálnímu minimu. Proto jsme zvolili raději kombinaci metody největšího spádu s Newtonovou metodou. Následnou inverzí funkce  $H(x)$  jsme dostali závislost výšky na barvě:

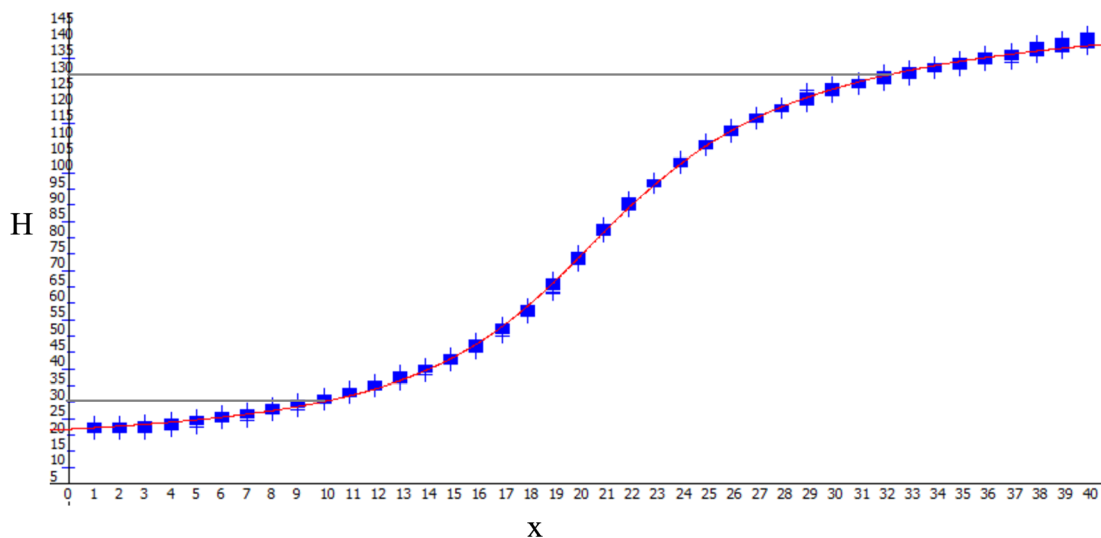
$$x = \frac{\tan\left(\pi\left[\frac{H-D}{C}\right]^{\frac{1}{E}} - \frac{\pi}{2}\right) - B}{A} \quad (3.2)$$



Obr. 3.1: Snímky zrcátka se vzájemným posunem výšky o 0,005 mm



Obr. 3.2: Závislost barvy na výšce pro dva různé pixely se stejným tónem barvy v pátém snímku

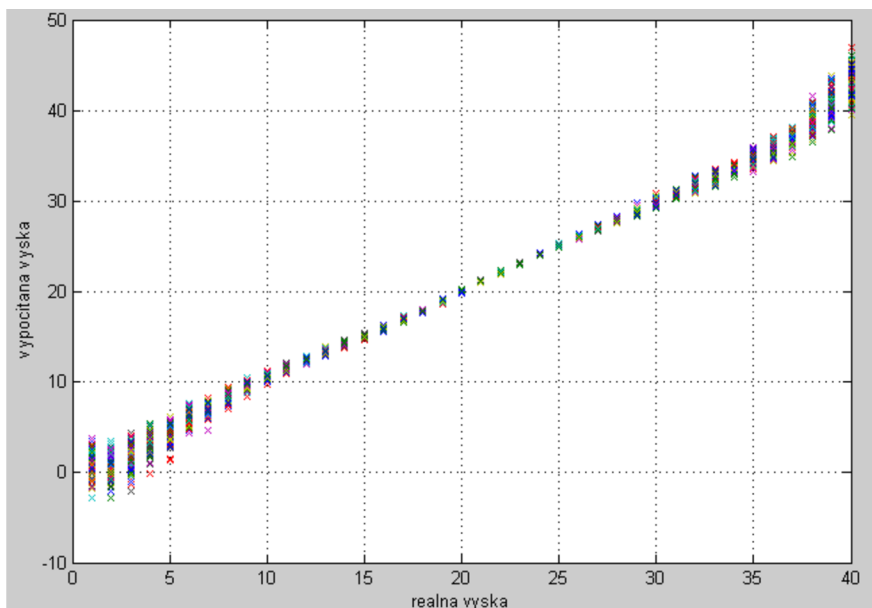


Obr. 3.3: Závislost barvy na výšce při volbě čtvercového okolí pixelu  $\mathcal{P}[240, 300]$  o velikosti 5

## 3.2 Zpracování obrazu

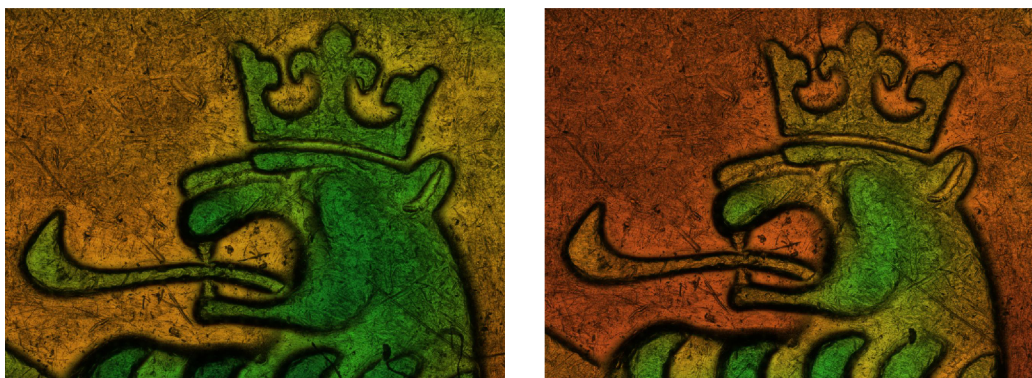
Jestliže známe závislost výšky na barvě, můžeme rekonstruovat povrch objektu z jediného snímku. My jsme použili snímek s měřítkem  $1 \text{ px} = 0,0018408425 \text{ mm}$ . Aby zrekonstruovaný povrch objektu byl co nejméně nejvhodnější, je třeba mít snímek s vhodnými tóny barvy.

Obrázek 3.4 zobrazuje závislost vypočtené výšky na reálné výšce. Z rozptylu vypočtených hodnot výšky je patrné, že by byl nejvhodnější takový snímek, který zahrnuje barevné tóny odpovídající snímekům 10 až 32, tj.  $H \in \langle 30^\circ, 125^\circ \rangle$ .



Obr. 3.4: Analýza odchylky skutečné a vypočítané výšky

Pro rekonstrukci povrchu jsme použili dva různé snímky téhož povrchu s různým intervalem barev (viz obrázek 3.5).



(a) Snímek č. 1 - ideální

(b) Snímek č. 2

Obr. 3.5: Snímky vhodné ke zpracování

Pro celý snímek jsme určili hodnotu tónu barvy  $H$  s předem určeným minimálním jasem. Pixely s nízkou jasovou intenzitou jsou zatíženy velkou barevnou chybou a proto je pro rekonstrukci výšky povrchu vylučujeme. Tyto pixely vznikají v místech, kde reliéf strmě stoupá, což má za následek odrazení světla mimo objektiv. Vznikly nám tedy části obrazu, kde není definována barva a tím i výška profilu.

Pomocí inverzní funkce 3.2 jsme dostali hodnoty relativní výšky, které jsme transformovali tak, aby minimální hodnota výšky byla rovna jedné. Nulové hodnoty výšky určují, že v těchto částech obrazu hodnotu výšky neznáme. To jsou pixely s nízkým jasem a pixely s tónem barvy  $H(x) \geq H(0)$ . Přibližnou představu o výškovém profilu je možné vidět na obrázku 3.6.

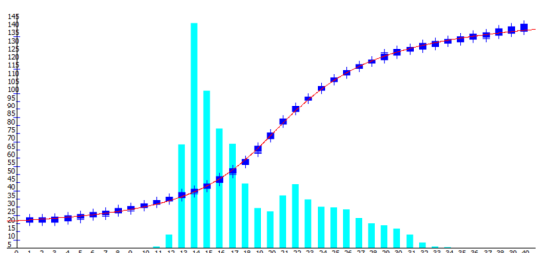




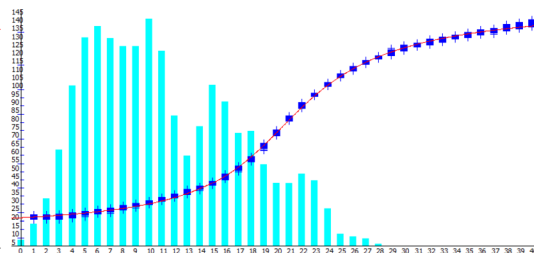
(a) Snímek č. 1 - jas větší než 0,1



(b) Snímek č. 2 - jas větší než 0,17



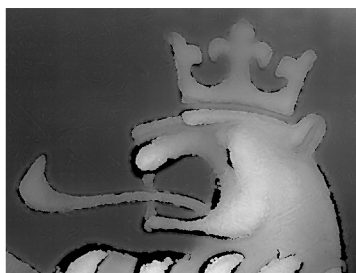
(c) Histogram pro snímek č. 1



(d) Histogram pro snímek č. 2

Obr. 3.6: Výškový profil s histogramem zjištěné výšky

K zjištění neznámých hodnot výšky bylo třeba použít vhodné numerické metody. My jsme použili metodu neúplné konvoluce s kruhovým gaussovským jádrem, která dává velmi dobré výsledky. Volbou  $\sigma$  v 2.11 se mění jednak vlastnost interpolace a zároveň i míra vyhlazení reliéfu povrchu. Metodu bylo třeba použít vícekrát nejdříve s malým jádrem s malým  $\sigma$ , abychom se zbavili menších neznámých částí, a následně s větším jádrem s větším  $\sigma$ , abychom dopočítali místa, kde jsou jasné hrany a velké „skoky“ výšky mezi sousedními pixely. Výškový profil po každé konvoluci jsou vidět na obrázcích 3.7 a 3.8.

(a) po 1. konvoluci s jádrem velikosti  $7 \times 7$  a  $\sigma = 2$ (b) po 2. konvoluci s jádrem velikosti  $7 \times 7$  a  $\sigma = 2$ (c) po 3. konvoluci s jádrem velikosti  $81 \times 81$  a  $\sigma = 8$ 

Obr. 3.7: Výškový profil snímku č.1



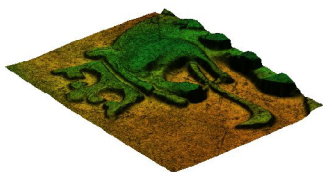
(a) po 1. konvoluci s jádrem velikosti  $7 \times 7$  a  $\sigma = 2$  (b) po 2. konvoluci s jádrem velikosti  $7 \times 7$  a  $\sigma = 2$  (c) po 3. konvoluci s jádrem velikosti  $81 \times 81$  a  $\sigma = 8$

Obr. 3.8: Výškový profil snímku č.2

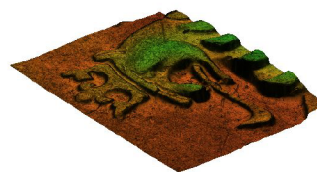
Přestože snímek č.2 zahrnoval barevné tóny nižší než označené jako ideální, dostali jsme dobré výsledky.

### 3.3 3D model

Pokud již známe relativní výšku ve všech pixelech obrazu, můžeme přejít k samotnému vykreslení pomocí knihovny OpenGL, s kterou se můžete blíže seznámit v bakalářské práci [5] a která využívá projektivního prostoru blíže popsaného v kapitole 1.1.



(a) Snímek č. 1

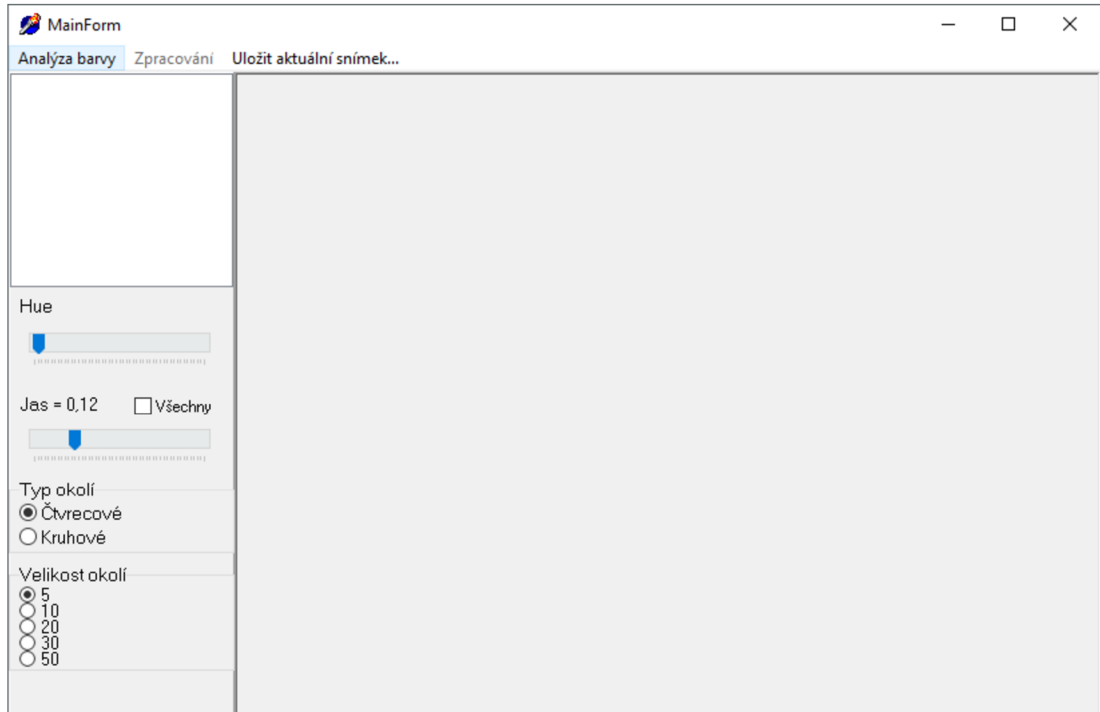


(b) Snímek č. 2

Obr. 3.9: 3D model - horizontální úhel  $230^\circ$ , vertikální  $35^\circ$  a vzdálenost 2

## 3.4 Aplikace

Tato kapitola obsahuje popis obsluhy vytvořené aplikace.



Obr. 3.10: Rozhraní aplikace

### 3.4.1 Zjištění závislosti výšky povrchu na barvě

#### Krok 1

Nejdříve z menu **Analýza barvy** vybereme položku **Otevřít snímky...** a v otvíracím dialogu vybereme všechny kalibrační snímky. Snímky musí být 16-bitové ve formátu TIF.

#### Krok 2

Nyní dvojklikem v listboxu vybereme jeden referenční snímek, čímž tento snímek zobrazíme. Klikem levého tlačítka na obrázek zjistíme tón barvy  $H$  pixelu (tzv. Hue) a polohu pixelů, které mají tento tón barvy nebo alespoň z intervalu kolem tohoto tónu, který si můžeme nastavit pomocí „šoupátka“ pod nápisem Hue, přičemž jeden dílek odpovídá  $\frac{1}{30}^\circ$ , a které mají hodnotu jasu větší než hodnota nastavená pomocí spodního „šoupátka“. Tyto pixely se ve snímku zobrazí jako bílé.

#### Krok 3

Stiskem pravého tlačítka v obrazu vybereme pixel s jeho okolím, ve kterém budeme zjišťovat závislost. Pokud bychom chtěli brát celé okolí, musíme vybrat položku Všechny,

v opačném případě je třeba, aby okolí obsahovalo, alespoň jeden bílý pixel a závislost se pak bude zjišťovat pouze v takto označených pixelech. Je možné nastavit velikost a tvar okolí.

#### Krok 4

Nyní z menu **Analýza barvy** vybereme položku **Graf**, což způsobí vytvoření formuláře s bodovým grafem závislosti barvy na výšce.

Nyní stiskneme tlačítko **Gradient** a následně **Newton**, čímž zjistíme funkční závislost výšky povrchu na barvě a tlačítkem **OK** tuto závislost potvrdíme.

### 3.4.2 Zpracování obrazu

Po tom, co zjistíme funkční závislost výšky povrchu na barvě, můžeme se pustit do zpracování snímků.

#### Krok 5

Z menu **Zpracování** vybereme položku **Otevřít...** a v otvíracím dialogu vybereme jeden snímek pro zpracování. Následně nastavíme minimální hodnotu jasu a z menu **Zpracování** vybereme položku **Výškový profil**, čímž se nám zobrazí histogram a výškový profil.

#### Krok 6

Nyní můžeme provést konvoluci, abychom zjistili výšku ve všech místech obrazu. Tzn. v menu **Zpracování** vybereme položku **Konvoluce**, čímž se zobrazí dialog s nastavením velikosti konvolučního jádra (tlačítko **Nastav velikost**) a  $\sigma$  pro gaussovské jádro (tlačítko **Gauss**). Po stisknutí tlačítka **OK** se provede konvoluce s vybraným jádrem a zobrazí se výškový profil.

### 3.4.3 3D modely

Pokud již známe výšku ve všech místech obrazu, můžeme provést vizualizaci.

#### Vizualizace v rámci aplikace

V menu **Zpracování** vybereme položku **3D model**, čímž se otevře formulář s vizualizací využívající OpenGL. Zde je možné pomocí nastavení úhlů dle 1.5 měnit pohled na povrch objektu.

#### Stereolitografický model (STL)

Kromě okamžité vizualizace v aplikaci, je možné vytvořit model, který lze následně vytisknout na 3D tiskárně. Tento model se ukládá do souboru s příponou **\*.stl**, který lze generovat výběrem položky **Export jako \*.stl...** v menu **Zpracování**.



## ZÁVĚR

Podářilo se nám popsat matematický aparát potřebný pro rekonstrukci povrchu s využitými metodami, jednak numerické metody pro zjištění závislosti výšky povrchu na barvě, dále metody zpracování obrazové informace a také aparát pro vizualizaci zrekonstruovaného povrchu.

Dále se nám podařilo vytvořit aplikaci ve vývojovém prostředí Delphi pro rekonstrukci povrchu objektu z 16-bitových snímků ve formátu TIF z konfokálního mikroskopu s hyperchromatickým objektivem. Aplikace je schopna z kalibračních snímků určit závislost výšky povrchu na barvě. K tomu posloužili metody vícerozměrné optimalizace, zejména metoda největšího spádu a Newtonova metoda. Po obdržení funkční závislosti mezi barvou pixelu a jeho výškou jsme následně z jednoho snímku zrekonstruovali povrch sledovaného objektu. K tomu jsme implemenovali metodu neúplné konvoluce s gaussovským jádrem. Dále jsme povrch vizualizovali pomocí knihovny OpenGL a dále byl vytvořen jeho stereolitografický model pro 3D tisk ve formátu STL. Jednotlivé kroky rekonstrukce lze sledovat v podobě histogramů, grafů a snímků výškových profilů, které je možné navíc uložit ve formátu BMP.

Aplikace není jednoúčelová, tj. pro konkrétní optickou soustavu, ale lze s ní zpracovávat snímky z jiných optických soustav konfokálního mikroskopu s hyperchromatickým objektivem za předpokladu, že máme pro danou soustavu vhodné kalibrační snímky.

Jak již v textu bylo uvedeno, záměrná barevná vada objektivu není homogenní, to způsobuje zkreslení výšky profilu. Tento nedostatek lze dále korigovat tak, že pro každou část obrazu se určí funkční závislosti barvy na výšce povrchu. Tato úloha může být předmětem dalšího vývoje numerických metod rekonstrukce povrchu.



## LITERATURA

- [1] ČADEK, Martin. *Lineární algebra a geometrie III* [online]. Brno, 2003 [cit. 2017-03-19]. Dostupné z: <http://www.math.muni.cz/~cadek/la3/SKRIPTA.pdf>
- [2] DRUCKMÜLLER, Miloslav. *Numerické metody analýzy obrazů*. (přednášky) Brno: VUT v Brně Fakulta strojního inženýrství, 2016.
- [3] ČERMÁK, Libor a Rudolf HLAVIČKA. *Numerické metody*. Vyd. 2. Brno: Akademické nakladatelství CERM, 2008. ISBN 978-80-214-3752-4.
- [4] HORÁK, Pavel a Josef JANYŠKA. *Analytická geometrie* [online]. Brno, 2009 [cit. 2017-03-16]. Dostupné z: <https://www.math.muni.cz/~janyska/AFPR-H.pdf>
- [5] JEDLIČKOVÁ, Barbora. *Modelování geometrických ploch*. Brno: Vysoké učení technické v Brně, Fakulta strojního inženýrství, 2015. 37 s. Vedoucí doc. Ing. Pavel Štarha, Ph.D.
- [6] KARÁSEK, Jiří a Ladislav SKULA. *Lineární algebra: teoretická část*. Brno: Akademické nakladatelství CERM, 2005. ISBN 80-214-3100-8.
- [7] KLAPKA, Jindřich. *Metody operačního výzkumu*. Vyd. 2. Brno: VUTIUM, 2001. ISBN 80-214-1839-7.
- [8] KLÍMA, Miloš. *Zpracování obrazové informace*. V Praze: České vysoké učení technické, 1996. ISBN 80-010-1436-3.
- [9] MARTIŠEK, Dalibor. 03-Projektivní prostor. In: *Počítačová geometrie a grafika: Přednáška* [online]. 22.11.2012 [cit. 2017-03-25]. Dostupné z: <http://mathonline.fme.vutbr.cz/Prednaska/sc-1245-sr-1-a-261/default.aspx>
- [10] MARTIŠEK, Dalibor. *Matematické principy grafických systémů*. Brno: Littera, 2002, 278 s. ISBN 80-857-6319-2.
- [11] PETRÁŇ, Mojmir. *Konfokální mikroskop s dvojitým řádkováním*. (přednáška) [online]. schůze Učené společnosti, 2014 [cit. 2017-05-16]. Dostupné z: [http://www.learned.cz/userfiles/pdf/prednasky-cleny-odborne/Petran-prednaska\\_2014.pdf](http://www.learned.cz/userfiles/pdf/prednasky-cleny-odborne/Petran-prednaska_2014.pdf)
- [12] ŠTARHA, Pavel. *Aplikace numerických metod zpracování obrazové informace*. Brno, 2015. Habilitační práce.
- [13] WIKIPEDIA, The Free Encyclopedia. *Barevný model* [online]. Poslední editace 31. 7. 2016 [cit. 2017-05-09]. Dostupné z: [https://cs.wikipedia.org/wiki/Barevn%C3%BD\\_model](https://cs.wikipedia.org/wiki/Barevn%C3%BD_model)

- [14] WIKIPEDIA, The Free Encyclopedia. *Barevný prostor* [online]. Poslední editace 3. 2. 2017 [cit. 2017-05-09]. Dostupné z: [https://cs.wikipedia.org/wiki/Barevn%C3%BD\\_prostor](https://cs.wikipedia.org/wiki/Barevn%C3%BD_prostor)
- [15] WILLIAM K. PRATT. *Digital Image Processing (Third Edition) PIKS Inside* [online]. 3rd ed. New York: Wiley-Interscience, 2001 [cit. cit. 2017-05-24]. ISBN 978-047-1221-326. Dostupné z: <http://onlinelibrary.wiley.com/book/10.1002/0471221325>
- [16] Zašlapané projekty: Mikroskop, který předběhl dobu. In: *Česká televize* [online]. 2009 [cit. 2017-05-22]. Dostupné z: <http://www.ceskatelevize.cz/porady/10209988352-zaslapane-projekty/409235100061018-mikroskop-ktery-predbeh1-dobu/>

## SEZNAM SYMBOLŮ A ZKRATEK

$\mathbf{u}, \mathbf{v}, \mathbf{w}/\mathbf{o}$	vektory/nulový vektor
$\mathbb{V} = (V, +)$	vektorový prostor
$L(\mathbf{v}_1, \dots, \mathbf{v}_n)$	lineární kombinace vektorů $\mathbf{v}_1, \dots, \mathbf{v}_n$
$\mathcal{B} = \langle \mathbf{v}_1, \dots, \mathbf{v}_n \rangle$	báze vektorového prostoru
$\langle \mathbf{u}, \mathbf{v} \rangle$	skalární součin vektorů $\mathbf{u}$ a $\mathbf{v}$
$\ \mathbf{v}\ $	norma vektoru $\mathbf{v}$
$\varphi$	úhel vektorů
$\mathcal{A}_n$	$n$ -rozměrný afinní prostor
$\mathcal{B}$	podprostor afinního prostoru
$\mathcal{R} = \langle O, \mathbf{v}_1, \dots, \mathbf{v}_n \rangle$	afinní repér (afinní souřadnicový systém)
$[a_1, \dots, a_n]$	souřadnice bodu $A$
$(u_1, \dots, u_n)$	souřadnice vektoru $\mathbf{u}$
$\mathbb{E}_n$	$n$ -rozměrný euklidovský prostor
$\mathbb{P}_n$	$n$ -rozměrný projektivní prostor
$\bar{A}$	geometrický bod projektivního prostoru
$\bar{\mathbb{E}}_n$	rozšířený $n$ -rozměrný euklidovský prostor
$\bar{A}_\infty$	nevlastní geometrický bod projektivního prostoru
$\bar{S}$	střed promítání
$h(x, y), g(x, y), h(x, y)$	funkce dvou proměnných
$\mathcal{I}_{m \times n}$	obrazová matice
$\mathcal{P}[i, j]$	pixel o souřadnicích $[i, j]$
$\mathcal{O}^{\mathcal{P}}(i, j, r)$	okolí pixelu $\mathcal{P}[i, j]$ o poloměru $r$
$\mathbf{G}$	konvoluční jádro



## SEZNAM PŘÍLOH

A Přiložené CD a 3D modely

50





## A PŘILOŽENÉ CD A 3D MODELY

Obsah přiloženého CD:

/ .....	kořenový adresář přiloženého CD
└─ Aplikace .....	aplikace se zdrojovým kódem
└─ Diplomova_prace.dpr	
└─ Diplomova_prace.dproj	
└─ Diplomova_prace.exe .....	spustitelná aplikace
└─ GraphUnit.dfm .....	formulář pro zobrazení grafu
└─ GraphUnit.pas	
└─ KonvKernelUnit.dfm .....	formulář nastavení konvolučního jádra
└─ KonvKernelUnit.pas	
└─ MainUnit.dfm .....	hlavní formulář
└─ MainUnit.pas .....	vlastní knihovna pro práci s maticemi
└─ MaticeUnit.pas	
└─ ModelUnit.dfm .....	formulář pro 3D vizualizaci pomocí OpenGL
└─ ModelUnit.pas	
└─ OpenTIFF.pas .....	knihovna pro práci s obrázky ve formátu TIF
└─ Snímky .....	snímky použité při rekonstrukci
└─ Kalibracni_snimky	
└─ Image_20170102_065.tif	
└─ Image_20170102_066.tif	
└─ :	
└─ Image_20170102_104.tif	
└─ Pro_rekonstrukci	
└─ Hlava_01_1304x976.tif	
└─ Hlava_02_1304x976.tif	
└─ Vysledky .....	Výsledky ze zpracování snímků
└─ 10_vysprof.bmp .....	Výškový profil (VP) - jas větší než 0,1
└─ 11_vysprof.bmp .....	VP po 1. konvoluci s jádrem o vel. $7 \times 7$ a $\sigma = 2$
└─ 12_vysprof.bmp .....	VP po 2. konv. s jádrem o vel. $7 \times 7$ a $\sigma = 2$
└─ 13_vysprof.bmp .....	VP po 3. konv. s jádrem o vel. $81 \times 81$ a $\sigma = 8$
└─ 14_3D.bmp .....	3D model - horizont. úhel $230^\circ$ , vert. úhel $35^\circ$ a vzdál. 2
└─ 15_Model.stl .....	Vygenerovaný stereolitografický model
└─ 20_vysprof.bmp .....	VP - jas větší než 0,17
└─ 21_vysprof.bmp .....	VP po 1. konv. s jádrem o vel. $7 \times 7$ a $\sigma = 2$
└─ 22_vysprof.bmp .....	VP po 2. konv. s jádrem o vel. $7 \times 7$ a $\sigma = 2$
└─ 23_vysprof.bmp .....	VP po 3. konv. s jádrem o vel. $81 \times 81$ a $\sigma = 8$
└─ 24_3D.bmp .....	3D model - horizont. úhel $230^\circ$ , vert. úhel $35^\circ$ a vzdál. 2
└─ 25_Model.stl .....	Vygenerovaný stereolitografický model

K tištěné verzi diplomové práce jsou mimo CD přiloženy také 3D modely zrekonstruovaných povrchů vytisknuté na 3D tiskárně z materiálu ABS a zvětšené na 2500% s výškou kroku 0,1 mm.