



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA STROJNÍHO INŽENÝRSTVÍ

FACULTY OF MECHANICAL ENGINEERING

ÚSTAV MATEMATIKY

INSTITUTE OF MATHEMATICS

**NUMERICKÉ METODY ANALÝZY OBRAZU ZAMĚŘENÉ
NA PROTÍNAJÍCÍ SE OBJEKTY**

NUMERICAL METHODS OF IMAGE ANALYSIS FOCUSED ON INTERSECTING OBJECTS

DIPLOMOVÁ PRÁCE

MASTER'S THESIS

AUTOR PRÁCE

AUTHOR

Bc. Juraj Weszter

VEDOUCÍ PRÁCE

SUPERVISOR

doc. Ing. Pavel Štarha, Ph.D.

BRNO 2021

Zadaní diplomové práce

Ústav:	Ústav matematiky
Student:	Bc. Juraj Weszter
Studijní program:	Aplikované vědy v inženýrství
Studijní obor:	Matematické inženýrství
Vedoucí práce:	doc. Ing. Pavel Štarha, Ph.D.
Akademický rok:	2020/21

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 analýzy obrazu zaměřené na protínající se objekty

Stručná charakteristika problematiky úkolu:

Velmi častým problémem při zpracování obrazové informace je vzájemné protínání se objektů. V tomto případě je segmentace velmi obtížná a vyžaduje individuální přístup pro vlastní analýzu dat.

Cíle diplomové práce:

Nastudovat základní metody zpracování obrazové informace a segmentaci dat.

Popsat použité numerické metody zpracování obrazové informace zaměřené na protínající se objekty.

Naprogramovat jednoúčelovou aplikaci pro dané zpracování dat.

Seznam doporučené literatury:

MARTIŠEK, Dalibor. Matematické principy grafických systémů. Brno: Littera, 2002, 278 s. ISBN 80-857-6319-2.

KLÍMA, Miloš. Zpracování obrazové informace. V Praze: České vysoké učení technické, 1996. ISBN 8001014363.

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 2020/21

V Brně, dne

L. S.

prof. RNDr. Josef Šlapal, CSc.
ředitel ústavu

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

Abstrakt

V práci je prezentovaný postup spracovania digitálnych snímok zachytávajúcích vlákna sive Nostoc za účelom merania ich dĺžky. Predstavený je algoritmus využívajúci Houghovu transformáciu na vyhodnotenie návaznosti vlákien v miestach ich priesečníkov. Algoritmus je demonštrovaný na vybraných snímkach, skúmané vlákna sú separované a vyhodnotené sú ich dĺžky. Súčasťou práce je softvér realizujúci prezentovaný algoritmus vo vývojovom prostredí Delphi.

Abstract

This thesis presents an image processing approach to estimating the length of cyanobacteria strands in digitally acquired images. An algorithm utilizing the Hough transform to determine strand continuity at strand intersections is presented. The algorithm is demonstrated on selected images, the examined strands are separated and their lengths are estimated. A Delphi implementation of the algorithm is included.

Ključové slová

digitálna krivka, pretínajúce sa objekty, Houghova transformácia, spracovanie obrazu

Keywords

digital curve, intersecting objects, Hough transform, image processing

WESZTER, Juraj. *Numerické metody analýzy obrazu zaměřené na protínající se objekty*. Brno, 2021. Dostupné také z: <https://www.vutbr.cz/studenti/zav-prace/detail/133865>. Diplomová práce. Vysoké učení technické v Brně, Fakulta strojního inženýrství, Ústav matematiky. 60 s. Vedoucí práce Pavel Štarha.

Diplomovú prácu som vypracoval sám, pod odborným vedením doc. Ing. Pavla Štarhu, Ph.D., s pomocou literárnych zdrojov uvedených v časti Literatúra.

Juraj Weszter

Za ústretovú pomoc, cenné rady, poradenstvo a všetok čas, ktorý mi venoval, ďakujem vedúcemu diplomovej práce, doc. Ing. Pavlovi Štarhovi, Ph.D.

Juraj Weszter

Obsah

Úvod	2
1 Matematický aparát	3
1.1 Binárne relácie	3
1.2 Metrický priestor	5
1.3 Konvolúcia	6
2 Vybrané metódy spracovania obrazu	8
2.1 Obraz a jeho vlastnosti	8
2.2 Lineárna filtrácia	11
2.3 Mediánová filtrácia	12
2.4 Prahovanie	13
2.5 Morfológické operácie	14
2.5.1 Dilatácia a erózia	15
2.5.2 Uzavretie a otvorenie	17
2.5.3 Hit-or-miss transformácia	18
2.5.4 Geodetická dilatácia	19
2.6 Flood fill	20
2.7 Houghova transformácia	21
3 Predspracovanie obrazových dát	25
3.1 Filtrácia	25
3.2 Segmentácia	26
3.3 Morfológická rekonštrukcia	27
3.4 Stenčovanie a čistenie	28
4 Analýza pretínajúcich sa objektov	30
4.1 Trasovanie	30
4.2 Estimácia dĺžky	38
5 Výsledky	41
Záver	43
Literatúra	43
A Snímky	46
B Opis vytvoreného softvéru	48

Úvod

Medzi často sa vyskytujúce problémy v obore analýzy obrazu patrí extrakcia vlastností objektov, ktoré sa v danom obraze nachádzajú. Na to, aby mohli byť tieto objekty analyzované, je najprv nutné ich v obraze správne identifikovať. V závislosti od skúmaných vlastností existuje na riešenie tohto problému viacero matematických metód – príkladom je Houghova transformácia na detekciu parametricky vyjadriteľných rovinných objektov, najčastejšie priamok. Situácia, kedy sa objekty v obraze prekrývajú alebo pretínajú, predstavuje podstatné sťaženie tejto úlohy. Vychádzajúc zo samotnej podstaty obrazových dát je tak často nutné metodiku obrazovej analýzy prispôbiť konkrétnej úlohe.

Predmetom diplomovej práce je numerická analýza prekrývajúcich sa objektov v obraze – analyzované sú snímky vláknitej sinice *Nostoc* zachytené mikroskopom, pričom vlastnosťou, ktorá má byť vyhodnotená, je ich dĺžka. Každá zo súboru spracovávaných snímok obsahuje niekoľko vlákien, ktoré sa v určitých miestach pretínajú. Rutinná aplikácia štandardných metód detekcie objektov v obraze nie je v tomto prípade postačujúca, nakoľko pred vyčíslením dĺžky vlákna je nutné dané vlákno separovať od vlákien ostatných.

V práci je predstavený algoritmus, ktorý separáciu vlákien realizuje. Houghova transformácia, ktorá je nástrojom na detekciu *priamok* v šedotónovom obraze, nachádza v algoritme nové uplatnenie – využitá je na *lokálnu aproximáciu „sklonu“ krivky* v miestach, kde sa táto krivka pretína s krivkou inou. Na základe tejto informácie sú následne krivky v obraze automaticky trasované, čo umožňuje ich separáciu odpovedajúcu vizuálnemu posúdeniu a vyhodnoteniu človekom. Okrem „sklonu“ využíva algoritmus niektoré historické informácie o správaní sa danej krivky v minulosti.

V kapitole 1 sú uvedené vybrané definície a vety týkajúce sa binárnych relácií, metrických priestorov a operácie konvolúcie, na základe ktorých je neskôr postavená matematická teória digitálneho obrazu a jeho spracovania.

V prvej časti kapitoly 2 je definovaný digitálny obraz a spomenuté sú jeho základné vlastnosti. V jej druhej časti sa nachádza matematický opis vybraných numerických metód spracovania obrazu, ktoré sú pri samotnej analýze použité.

Kapitola 3 opisuje postup predspracovania obrazových dát a ich prípravy na proces separácie objektov, ktoré zachytávajú. Uplatnené sú postupy a metódy z kapitoly 2 – snímky boli lineárne filtrované, segmentované a upravované vybranými morfológickými operáciami a numerickými metódami.

V kapitole 4 je predstavený samotný algoritmus, ktorý vykonáva separáciu pretínajúcich sa kriviek v binárnom obraze. Opis algoritmu je doprevádzaný ukážkou jeho krokov na jednej zo spracovávaných snímok. Uvedená je metodika estimácie dĺžky separovaných kriviek.

Výsledky práce sú zhrnuté v kapitole 5. Dĺžky vlákien siníc nachádzajúcich sa v spracovávaných snímkach sú uvedené v tabuľke a zhodnotená je funkčnosť algoritmu.

1 Matematický aparát

V nematematickom chápaní obraz vidíme ako súbor objektov, o ktorých môžeme vysloviť úsudok, že pozostávajú z bodov. Analýza, ktorá má byť vykonaná, si vyžaduje postup inverzný. Obraz už vopred považujeme za súbor bodov a až skúmajúc ich vlastnosti nachádzame vzťahy určujúce to, ktoré body sú súčasťou jedného objektu. Na matematickú formalizáciu týchto vzťahov potrebujeme predovšetkým aparát binárnych relácií a niektoré pojmy súvisiace s metrickými priestormi. Nasledujúce definície je možné považovať za všeobecne známe, ich formulácia však založená na monografii [1] a prednáškach [2], [3]. Podkapitola o konvolúcii vychádza z [4].

1.1 Binárne relácie

Definícia 1.1.1 (Karteziánsky súčin). Nech A, B sú ľubovoľné množiny. Množinu

$$A \times B := \{(a, b) \mid a \in A \wedge b \in B\}$$

nazveme *karteziánskym* súčinom množín A a B .

Definícia 1.1.2 (Binárna relácia). Nech A, B sú ľubovoľné množiny. *Binárnou reláciou* medzi množinami A a B nazveme ľubovoľnú podmnožinu

$$R \subseteq A \times B.$$

O prvkoch $a \in A, b \in B$ povieme, že sú v relácii R , a zapíšeme $a \sim_R b$, ak $(a, b) \in R$. V opačnom prípade zapíšeme $a \not\sim_R b$. Ak platí $A = B$, o R povieme, že je A binárnou reláciou *na množine* A .

Na základe toho, ako je relácia na množine určená, môžeme hovoriť o tom, či splňuje alebo nesplňuje nasledujúce vlastnosti.

Definícia 1.1.3 (Vlastnosti relácie). Nech R je reláciou na A . Povieme, že R je

- (i) *reflexívna*, ak platí $a \sim_R a$ pre všetky $a \in A$,
- (ii) *ireflexívna*, ak platí $a \not\sim_R a$ pre všetky $a \in A$,
- (iii) *symetrická*, ak platí $a_1 \sim_R a_2 \implies a_2 \sim_R a_1$ pre všetky $a_1, a_2 \in A$,
- (iv) *antisymetrická*, ak platí $a_1 \sim_R a_2 \wedge a_2 \sim_R a_1 \implies a_1 = a_2$ pre všetky $a_1, a_2 \in A$,
- (v) *asymetrická*, ak platí $a_1 \sim_R a_2 \implies a_2 \not\sim_R a_1$ pre všetky $a_1, a_2 \in A$,
- (vi) *tranzitívna*, ak platí $a_1 \sim_R a_2 \wedge a_2 \sim_R a_3 \implies a_1 \sim_R a_3$ pre všetky $a_1, a_2, a_3 \in A$.

Významným typom relácie na množine je relácia *ekvivalencie*, ktorá nám umožňuje danú množinu na základe určitého vzťahu medzi jej prvkami rozdeliť. Pri analýze v kapitole 2 bude práve vhodne definovaná relácia ekvivalencie poskytovať spôsob, ktorý umožní v obraze rozlišovať objekty od pozadia a objekty medzi sebou.

Definícia 1.1.4 (Rozklad množiny). Nech $A \neq \emptyset$ je ľubovoľná množina. Množinu $\mathcal{M} \subseteq \mathcal{P}(A)$ splňujúcu

- (i) $\emptyset \notin \mathcal{M}$,
- (ii) $M_1 \cap M_2 = \emptyset$ pre všetky $M_1, M_2 \in \mathcal{M}$ také, že $M_1 \neq M_2$,
- (iii) $\bigcup_{M \in \mathcal{M}} M = A$,

nazveme *rozkladom množiny A*. Množiny $M \in \mathcal{M}$ nazveme *triedami rozkladu*.

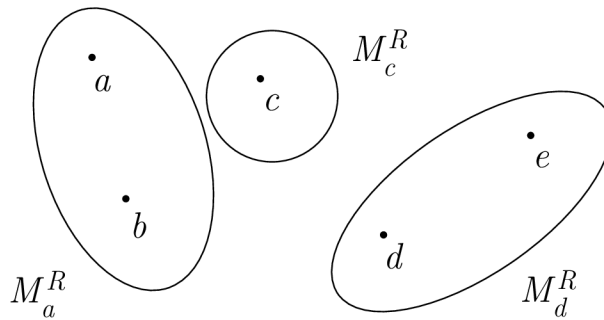
Definícia 1.1.5 (Ekvivalencia). Binárnu reláciu na A , ktorá je reflexívna, symetrická a tranzitívna, nazveme reláciou *ekvivalencie* na A .

Veta 1.1.1. Nech R je relácie ekvivalencie na A . Množina

$$\mathcal{M}_R := \{M_a^R \mid a \in A\}, \quad \text{kde} \quad M_a^R = \{a' \in A \mid a \sim_R a'\},$$

je rozkladom množiny A .

Relácia ekvivalencie definovaná na množine A tak túto množinu rozkladá na systém navzájom disjunktných podmnožín, ktorých zjednotením je celá množina A .



Obr. 1.1: Rozklad množiny $M = \{a, b, c, d, e\}$ binárnou reláciou ekvivalencie $R = \{(a,a), (b,b), (c,c), (d,d), (e,e), (a,b), (b,a), (d,e), (e,d)\}$

Nakoľko relácie sú definované ako množiny usporiadaných dvojíc, pomocou klasických množinových operácií dokážeme zaviesť aj operácie s reláciami.

Definícia 1.1.6 (Operácie s reláciami). Nech R a S sú relácie medzi množinami A a B a nech T je relácia medzi množinami B a C .

- (i) *Zjednotením* relácií R a S nazveme reláciu

$$R \cup S := \{(a, b) \in A \times B : aRb \vee aSb\}.$$

- (ii) *Prienikom* relácií R a S nazveme reláciu

$$R \cap S := \{(a, b) \in A \times B : aRb \wedge aSb\}.$$

- (iii) *Zložením* relácií R a S nazveme reláciu

$$T \circ R := \{(a, c) \in A \times C : \exists b \in B \text{ také, že } aRb \wedge bTc\}.$$

(iv) *Inverznou* reláciou k relácii R nazveme reláciu

$$R^{-1} := \{(b, a) \in B \times A : aRb\}.$$

1.2 Metrický priestor

Pri spracovaní obrazu sa nezaobídeme bez konceptu vzdialenosti, ktorého matematickou formalizáciou je *metrika*. Aparát metrických priestorov nám mimo iné umožňuje používať pojem susednosti pixelov, ktorého znalosť je nutná už aj pri základných postupoch obrazovej analýzy.

Definícia 1.2.1 (Metrika a metrický priestor). Nech $M \neq \emptyset$. Zobrazenie $\varrho : M \times M \rightarrow [0, \infty)$ splňuje

$$(i) \quad \varrho(x, y) = 0 \iff x = y$$

$$(ii) \quad \varrho(x, y) = \varrho(y, x)$$

$$(iii) \quad \varrho(x, z) \leq \varrho(x, y) + \varrho(y, z)$$

pre všetky $x, y, z \in M$ nazveme *metrikou* na M . Dvojicu (M, ϱ) nazveme *metrickým priestorom*.

Metrika môže byť všeobecne ľubovoľná, ak splňuje vyššie uvedené axiómy. Axióm (i) nazývame axióm *totožnosti* – hovoríme, že metrika rozlišuje body. Axióm (ii) je axióm *symetrie* a axióm (iii) je nazývaný *trojuholníková nerovnosť*. Medzi najznámejšie a najbežejšie používané metriky patrí *euklidovská* metrika, ktorá formalizuje intuitívne chápaný pojem vzdialenosti v reálnej rovine. V kontexte digitálneho obrazu sa však nepohybujeme v reálnej rovine, a vhodnejšiou (no nie jednoznačnou) voľbou sa ukazujú byť metriky *súčtová* a *maximová*.

Definícia 1.2.2 (Vybrané metriky). Nech $M = \mathbb{R}^n, n \in \mathbb{N}$. Metriku definovanú

$$\varrho_1(x, y) := \sum_{k=1}^n |x_k - y_k| \tag{1.1}$$

nazveme *súčtovou* (manhattanskou) metrikou. Metriku definovanú

$$\varrho_2(x, y) := \left(\sum_{k=1}^n (x_k - y_k)^2 \right)^{\frac{1}{2}} \tag{1.2}$$

nazveme *euklidovskou* metrikou a metriku definovanú

$$\varrho_\infty(x, y) := \max\{|x_1 - y_1|, |x_2 - y_2|, \dots, |x_n - y_n|\} \tag{1.3}$$

nazveme *maximálnou* (*Chebyshevovou*) metrikou.

V predchádzajúcich definíciách predpokladáme, že M je podmnožinou \mathbb{R}^n . Poznamenajme však, že nakoľko platí $\mathbb{Z}^n \subset \mathbb{R}^n$, metriku môžeme používať aj v n -rozmernom celočíselnom obore, alebo jeho ľubovoľnej neprázdnej podmnožine.

Definícia 1.2.3 (Guľa). Nech (M, ϱ) je metrický priestor, $x_0 \in M$ a r je kladná reálna konštanta. Množinu

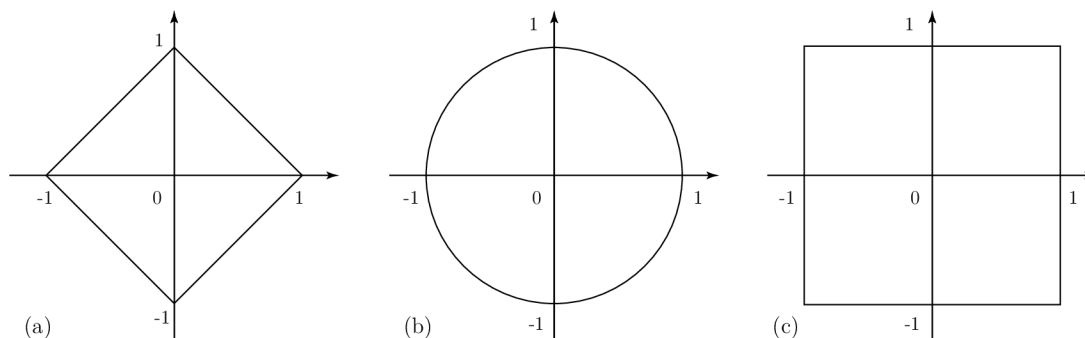
$$B_r(x_0) := \{x \in M : \varrho(x_0, x) < r\} \quad (1.4)$$

nazveme *otvorenou guľou* so stredom v bode x_0 a polomerom r . Množinu

$$B_r[x_0] := \{x \in M : \varrho(x_0, x) \leq r\} \quad (1.5)$$

nazveme *uzravretou guľou* so stredom v bode x_0 a polomerom r . Špeciálne, guľu s polomerom $r = 1$ nazveme *jednotkovou guľou*.

Poznámka 1.2.1. Pre $\epsilon \in \mathbb{R}^+$ nazveme $O_\epsilon(x_0) := B_\epsilon(x_0)$ epsilonovým okolím (ϵ -okolím) bodu x_0 .



Obr. 1.2: Jednotková guľa v $\mathbb{R} \times \mathbb{R}$ so stredom v bode $(0, 0)$ pri použití metriky (a) ϱ_1 (b) ϱ_2 (c) ϱ_∞

1.3 Konvolúcia

Definícia 1.3.1 (Konvolúcia). Nech $f, h : \mathbb{R} \rightarrow \mathbb{R}$. Funkciu

$$(f * g)(t) := \int_{-\infty}^{\infty} f(\tau) h(t - \tau) d\tau \quad (1.6)$$

nazveme *konvolúciou* funkcie f s konvolučným jadrom h .

Veta 1.3.1 (Vlastnosti konvolúcie). Nech f, g , a h sú reálne funkcie reálnej premennej a nech a je reálna konštanta. Pre konvolúciu platí

- (i) $f * g = g * f$,
- (ii) $f * (g * h) = (f * g) * h$,
- (iii) $f * (g + h) = (f * g) + (f * h)$,
- (iv) $a(f * g) = (af) * g = f * (ag)$.

Komutativita nám potom umožňuje v definícii 1.3.1 použiť ekvivalentný zápis

$$(f * g)(x, y) = \int_{-\infty}^{\infty} f(t - \tau) h(t) d\tau. \quad (1.7)$$

Aby konvolúcia našla svoje uplatnenie v obore spracovania digitálneho obrazu, je nutné ju najprv zovšeobecniť pre funkcie dvoch premenných.

Definícia 1.3.2 (Konvolúcia funkcie dvoch premenných). Nech $f, h : \mathbb{R} \times \mathbb{R} \rightarrow \mathbb{R}$. Funkciu

$$(f * g)(t) := \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(a, b) h(x - a, y - b) da db = \quad (1.8)$$

$$= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x - a, y - b) h(a, b) da db. \quad (1.9)$$

nazveme *konvolúciou* funkcie f s konvolučným jadrom h .

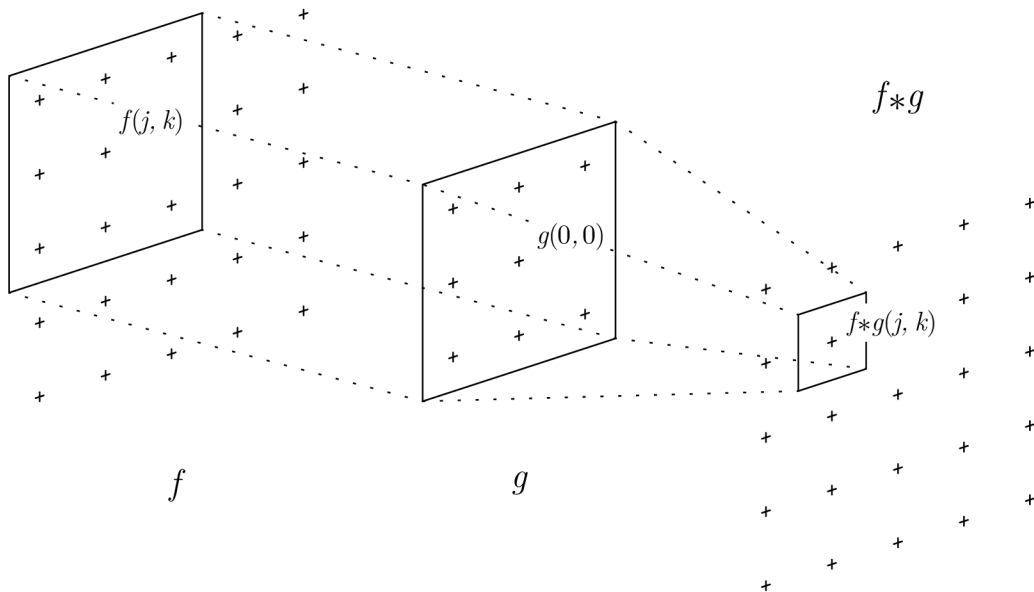
Argumentom takto definovanej konvolúcie je reálna funkcia dvoch reálnych premenných. V nasledujúcej kapitole však zavedieme digitálny obraz ako zobrazenie s celočíselným definičným oborom. Budeme preto potrebovať *diskrétnu* konvolúciu, ktorú získame nahradením integrálov definícii 1.3.2 konečnými sumami.

Definícia 1.3.3 (Diskrétna konvolúcia funkcie dvoch premenných). Nech $f : \mathbb{Z} \times \mathbb{Z} \rightarrow \mathbb{R}$, $g : \mathbb{Z} \times \mathbb{Z} \rightarrow \mathbb{R}$. Funkciu definovanú

$$(f * g)(j, k) := \sum_m \sum_n f(j - m, k - n) g(m, n) \quad (1.10)$$

nazveme *diskrétnou konvolúciou* funkcie f s konvolučným jadrom (maskou) g .

Konvolúcia je základom procesu *lineárnej filtrácie* obrazu, ktorá bude opísaná v časti 2.2.



Obr. 1.3: Diskrétna konvolúcia

2 Vybrané metódy spracovania obrazu

Z matematického pohľadu obrazom rozumieme zobrazenie priradujúce bodom svojho definičného oboru, nazývaných *pixelmi*, jasovú hodnotu. Toto zobrazenie môže byť všeobecne definované na akejkoľvek podmnožine reálnej roviny. Nakoľko však cieľom práce je spracovanie obrazu *digitálneho* a samotný proces digitalizácie nemá na uvedené realizované postupy dopad, budeme v nasledujúcom texte uvažovať výhradne obraz digitálny, a teda už v jeho definícii budeme predpokladať konečnosť jeho definičného oboru. Obsah tejto kapitoly vychádza primárne z literárnych zdrojov [4], [5], [6] a [7]. Sekcia 2.5 týkajúca sa matematickej morfológie je založená na monografiách [8] a [12]. Opis Houghovej transformácie v časti 2.7 sa opiera o vedomosti vychádzajúce z [11] a [14]. V neposlednej rade sú využité niektoré poznatky z [13].

2.1 Obraz a jeho vlastnosti

Definícia 2.1.1 (Obraz). Nech $M = \{1, 2, \dots, h\} \times \{1, 2, \dots, w\}$, kde $h, w \in \mathbb{N}$ a nech $R = \{0, 1, \dots, 2^k - 1\}$, kde $k \in \mathbb{N}$. Šedotónovým *obrazom* (stručne *obrazom*) o rozmeroch $h \times w$ a s *bitovou hĺbkou* k potom nazveme zobrazenie

$$i : M \rightarrow R.$$

Hodnotu $i(p)$ pixelu $p = (j, k)$ nazveme *jasovou* alebo *pixelovou* hodnotou pixelu p . Špeciálnym prípadom obrazu je *binárny obraz*, ktorého obor hodnôt R obsahuje práve dva prvky, spravidla

$$R = \{0, 1\}.$$

Niekedy ďalej býva vhodné obraz zapisovať v maticovom tvare, ktorý odpovedá rozmiestneniu pixelov v rovine a je daný predpisom

$$\mathbf{I} = [i_{jk}], \quad \text{kde } i_{jk} = i(j, k).$$

Konečnosť oboru hodnôt nám dovoľuje pre obraz i_1 zdefinovať obraz nový, ktorý je k i_2 v istom zmysle opačný. Takýto obraz je nazývaný *komplement* obrazu i_1 .

Definícia 2.1.2 (Komplement obrazu). Nech i_1 a i_2 sú obrazy o rozmeroch $h \times w$ s bitovou hĺbkou k . Obraz i_2 nazveme komplementom obrazu i_1 , ak

$$i_2(j, k) = (2^k - 1) - i_1(j, k) \quad \text{pre všetky } j = 1, \dots, w, \quad k = 1, \dots, h.$$

Poznámka 2.1.1. Binárna relácia daná

$$p \sim q \Leftrightarrow i(p) = i(q) \tag{2.1}$$

na množine M binárneho obrazu $i : M \rightarrow \{0, 1\}$ je reláciou ekvivalencie a podľa vety 1.1.1 tak rozkladá M na podmnožiny

$$\begin{aligned} M_O &= \{p \in M \mid i(p) = 1\}, \\ M_B &= \{p \in M \mid i(p) = 0\}, \end{aligned}$$

0	0	0	0	0	0	0
0	0	1	0	1	1	0
0	1	1	0	1	0	0
0	1	1	0	0	0	0
0	1	1	1	0	0	0

1	1	1	1	1	1	1
1	1	0	1	0	0	1
1	0	0	1	0	1	1
1	0	0	1	1	1	1
1	0	0	0	1	1	1

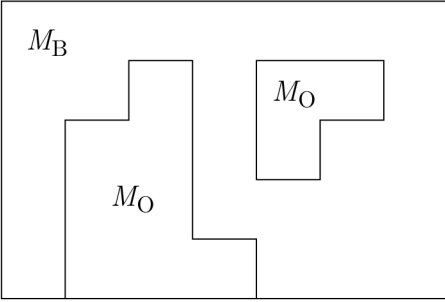
Obr. 2.1: Binárny obraz (vľavo) a jeho komplement (vpravo).

pričom

$$M_O \cap M_B = \emptyset, \quad M_O \cup M_B = M.$$

Zobrazenie i je zároveň charakteristickou funkciou množiny M_O . Potom je zrejmé, že ak poznáme definičný obor binárneho obrazu, množina M_O daný obraz jednoznačne určuje. Pixely $p \in M_O$ bývajú v kontexte rozpoznávania objektov nazývané *objektové pixely*, a $q \in M_B$ *pixely pozadia*. Ak potom hovoríme o objektoch v obraze, resp. pozadí obrazu, máme na mysli jeho objektové pixely, resp. pixely pozadia. V prípade, že potrebujeme v obraze rozlišovať nie len objekty od pozadia, ale aj objekty medzi sebou, musíme zadať pojem susednosti pixelov.

0	0	0	0	0	0	0
0	0	1	0	1	1	0
0	1	1	0	1	0	0
0	1	1	0	0	0	0
0	1	1	1	0	0	0



Obr. 2.2: Binárny obraz (vľavo) a jeho rozklad reláciou ekvivalencie 2.1 (vpravo).

Definícia 2.1.3 (Susednosť pixelov). Nech η je symetrická binárna relácia na M . O pixeloch p a q povieme, že sú η -susedné, ak

$$(p, q) \in \eta.$$

Definícia 2.1.4 (Cesta). Nech $A \subseteq M$, $n \in \mathbb{N}$. Postupnosť pixelov

$$p = p_0, \dots, p_n = q, \quad \text{kde } p_k \in A \text{ pre } k = 0, \dots, n,$$

nazveme η -cestou (*digitálnou krivkou*) v A medzi p a q , ak

$$p_{k-1} \text{ a } p_k \text{ sú } \eta\text{-susedné pre všetky } k = 1, \dots, n.$$

Prirodzené číslo n nazveme *dĺžkou cesty* (digitálnej krivky). Ďalej povieme, že táto cesta je *jednoduchá*, ak $p_k \neq p_l$ pre všetky indexy $k, l \in \{1, \dots, n\}$, $k \neq l$.

Definícia 2.1.5 (Spojené pixely). O pixeloch $p, q \in A \subseteq M$ povieme, že sú η -spojené v A , ak medzi p a q existuje η -cesta v A .

Definícia 2.1.6 (Súvislá podmnožina). Nech $A \subseteq M$, $A \neq \emptyset$. O množine A povieme, že je η -súvislou podmnožinou, ak ľubovoľné $p, q \in A$ sú η -spojené v A .

Nakoľko definícia 2.1.4 pripúšťa triviálne cesty nulovej dĺžky, binárna relácia „byť η -spojené v A “ je reflexívna. Navyše, už v definícii 2.1.3 predpokladáme symetriu. Ďalej je ľahké vidieť, že táto relácia je tranzitívna – ak existuje η -cesta medzi p a q a existuje η -cesta medzi q a r , potom postupnosť

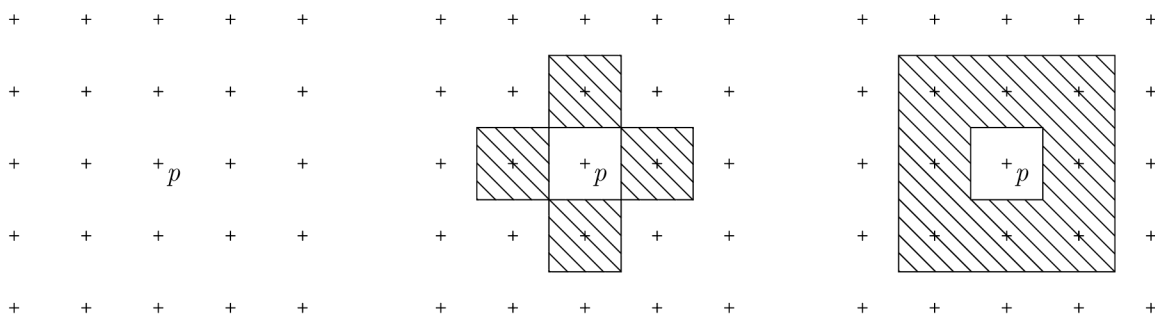
$$p, \dots, q, \dots, r$$

je η -cestou medzi p a r . Binárna relácia „byť η -spojené v A “ je tak reláciou ekvivalencie a podľa vety 1.1.1 množinu A rozkladá na η -súvislé podmnožiny, ktoré nazveme η -komponenty.

Relácia η môže byť zavedená rôzne – stačí, že splňuje predpoklady definície 2.1.3. V práci však budeme rozlišovať len dve, a to

$$\begin{aligned} (p, q) \in \eta_4 & \text{ práve vtedy, keď } p \neq q \wedge \varrho_1(p, q) \leq 1, \\ (p, q) \in \eta_8 & \text{ práve vtedy, keď } \varrho_\infty(p, q) = 1. \end{aligned}$$

kde ϱ_1 a ϱ_∞ sú metriky zavedené v časti 1.2. Ľahko si všimneme, že definičný obor obrazu spolu s jednou z metrick $\varrho_1, \varrho_\infty$ tvorí metrický priestor.



Obr. 2.3: Pixel p , množina $\mathcal{N}_4(p)$ a množina $\mathcal{N}_8(p)$.

S cieľom zjednodušenia budeme namiesto η_4 -susednosti, resp. η_8 -susednosti používať výraz 4-susednosť, resp. 8-susednosť. Množinu 4-susedných, resp. 8-susedných pixelov pixelu p obrazu $i : M \rightarrow R$ označme

$$\mathcal{N}_i^4(p) := \{q \in M \mid (p, q) \in \eta_4\} \quad \text{resp.} \quad \mathcal{N}_i^8(p) := \{q \in M \mid (p, q) \in \eta_8\}.$$

Ďalej označme $\mathcal{O}_i(p)$ množinu tých 8-susedných pixelov pixelu p obrazu $i : M \rightarrow \{0, 1\}$, ktoré sú objektové, t. j.

$$\mathcal{O}_i(p) := \{q \in \mathcal{N}_i^8(p) \mid i(q) = 1\}. \quad (2.2)$$

2.2 Lineárna filtrácia

Nakoľko obraz je podľa definície 2.1.1 funkciou dvoch celočíselých premenných, môžeme naň aplikovať operáciu diskkrétnej konvolúcie dvoch premenných definovanej v časti 1.3. Túto operáciu nazývame *lineárna filtrácia* alebo *lineárny filter*.

Definícia 2.2.1 (Lineárna filtrácia). Nech i_1 je obraz s rozmermi $h \times w$ a g je zobrazenie

$$\{-a, \dots, 0, \dots, a\} \times \{-b, \dots, 0, \dots, b\} \rightarrow \mathbb{Z}, \quad a, b \in \mathbb{N}.$$

Obraz i_2 s rozmermi $h \times w$ a pixelovými hodnotami

$$i_2(j, k) = \sum_{s=-a}^a \sum_{t=-b}^b g(s, t) i(j + s, k + t)$$

nazeveme *lineárnou filtráciou* obrazu i_1 s konvolučným jadrom g . Konvolučné jadro môžeme zapísať v tvare $(2a + 1) \times (2b + 1)$ matice $\mathbf{G} = [g_{jk}]$, kde

$$g_{jk} = g(j - a - 1, k - b - 1).$$

Matica \mathbf{G} tak má nepárny počet stĺpcov, nepárny počet riadkov a prvok, ktorý sa v nej nachádza „uprostred“, odpovedá hodnote $g(0, 0)$. Na základe výberu konvolučného jadra rozlišujeme dva významné typy lineárnych filtrov:

- (i) filter typu *horná priepusť*,
- (ii) filter typu *dolná priepusť*.

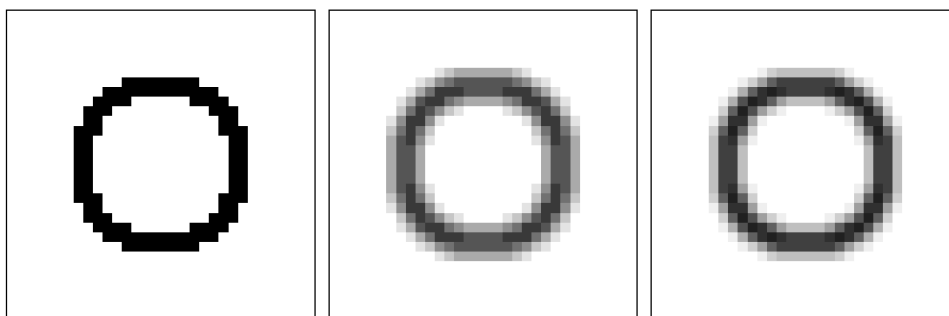
Filtre typu dolná priepusť sú základnom väčšiny metód vyhladzovania obrazu. Nahradením hodnoty pixelu (j, k) vhodným váženým priemerom hodnôt pixelov z jeho okolia dochádza k redukcii rozdielu v jasovej hodnote medzi susednými pixelmi. Použitím filtru typu dolná priepusť dochádza v obraze k potlačeniu vysokých frekvencií, zatiaľ čo frekvencie nízky sú zachované. Váhy, ktoré sú pri výpočte priemeru použité, sú dané prvkami konvolučnej matice. Voľbou

$$\mathbf{G} = \mathbf{G}_{n \times n} = \frac{1}{g_{11} + g_{12} + \dots + g_{nn}} \begin{bmatrix} g_{11} & g_{12} & \cdots & g_{1n} \\ g_{21} & g_{22} & \cdots & g_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ g_{n1} & g_{n2} & \cdots & g_{nn} \end{bmatrix}, \quad \text{kde } g_{ij} = 1 \quad \text{pre všetky } i, j,$$

dostávame $n \times n$ filter *kľavého priemeru*. Ak za prvky g_{ij} zvolíme hodnoty aproximujúce dvojrozmernú Gaussovu funkciu,

$$\mathbf{G}_{3 \times 3} = \frac{1}{16} \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}, \quad \mathbf{G}_{5 \times 5} = \frac{1}{256} \begin{bmatrix} 1 & 4 & 6 & 4 & 1 \\ 4 & 16 & 24 & 16 & 4 \\ 6 & 24 & 36 & 24 & 6 \\ 4 & 16 & 24 & 16 & 4 \\ 1 & 4 & 6 & 4 & 1 \end{bmatrix},$$

dostávame *Gaussov filter*.



Obr. 2.4: Pôvodný obraz, jeho úprava filtrom kľavého priemeru 3×3 a Gaussovým filtrom 5×5 .

Naopak, filtre typu horná priepusť obraz zostrujú. Pixelová hodnota pixelu (j,k) je opäť nahradená váženým priemerom hodnôt pixelov z jeho okolia – váhy sú však volené tak, aby sa ich jasový rozdiel zvýšil. Dochádza tak k potlačeniu nízkych frekvencií, zatiaľ čo vysoké frekvencie sú zachované. Príkladom filtru typu dolná priepusť je lineárny filter s konvolučným jadrom

$$\mathbf{G}_{3 \times 3} = \frac{1}{9} \begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}.$$

Filtre typu dolná priepusť nachádzajú využitie pri potlačovaní šumu v obraze. Na druhú stranu, filtre typu horná priepusť síce obraz doostria, amplifikujú však aj šum, ktorý sa v nich nachádza.

2.3 Mediánová filtrácia

Podobne ako filter lineárny, mediánový filter upravuje jasovú hodnotu pixelu $p = (j, k)$ obrazu i_1 na základe jasových hodnôt pixelov z jeho okolia. Namiesto váženého priemeru týchto hodnôt je však použitý ich medián, čím je potlačený nežiadúci vplyv impulzného

šumu. Ak za okolie pixelu p považujeme množinu jeho 8-susedných pixelov, po filtrácii dostávame obraz i_2 , ktorého jasové hodnoty sú

$$i_2(p) := \text{median} \left\{ \begin{array}{l} i(j-1, k-1), \quad i(j-1, k), \quad i(j-1, k+1), \\ i(j, k-1), \quad \quad \quad i(j, k+1), \\ i(j+1, k-1), \quad i(j+1, k), \quad i(j+1, k+1) \end{array} \right\}. \quad (2.3)$$

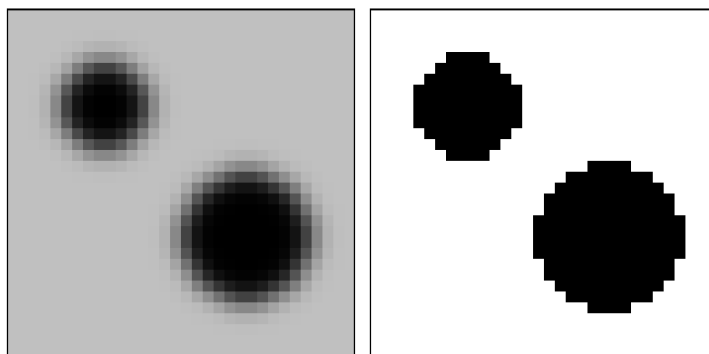
2.4 Prahovanie

V časti 2.1 bolo ukázané, že ak pracujeme s binárnym obrazom, dokážeme jeho definičný obor bez veľkej snahy rozložiť na dve triedy – množinu objektových pixelov a množinu pixelov pozadia. Ak teda chceme rozlišovať objekty od pozadia v obraze šedotónovom, stačí nám tento obraz vhodným spôsobom previesť na obraz binárny. Jedným z najjednoduchších spôsobov, ako tento prevod docieľiť, je *prahovanie*.

Prepokladajme, že obraz zachytáva, voľne povedané, „svetlé“ objekty na „tmavom“ pozadí. Proces prahovania obrazu $i_1 : M \rightarrow R$ potom spočíva vo vytvorení binárneho obrazu

$$i_2 : M \rightarrow \{0, 1\}, \quad \text{kde} \quad i_2(j, k) := \begin{cases} 1 & \text{ak} \quad i_1(j, k) < T, \\ 0 & \text{ak} \quad i_1(j, k) \geq T, \end{cases} \quad (2.4)$$

kde $T \in R$ je *prahová hodnota*. Pixely binárneho obrazu s hodnotou 1 sú potom objektovými a pixely s hodnotou 0 sú pixelmi pozadia. Hodnotu T dokážeme určiť napr. vizuálnym posúdením histogramu obrazu – za platnosti predpokladu vzájomne rôzneho jasov objektov a pozadia je histogram obrazu *bimodálny* a prirodzenou voľbou pre prah T je tak lokálne minimum nachádzajúce sa medzi vrcholmi histogramu.



Obr. 2.5: Pôvodný obraz a binárny obraz získaný jeho prahovaním.

V publikácii [9] je predstavený iteračný výpočet, ktorým je proces voľby prahovej hodnoty T možné automatizovať:

- (i) V kroku $t = 0$ zvolíme počiatočnú hodnotu $T^{(0)} = \frac{1}{hw} \sum_{j=1}^h \sum_{k=1}^w i_1(j, k)$.
- (ii) Prahovaním zostrojíme množinu $M_O^{(t)}$ objektových pixelov obrazu a množinu $M_B^{(t)}$ pixelov pozadia.

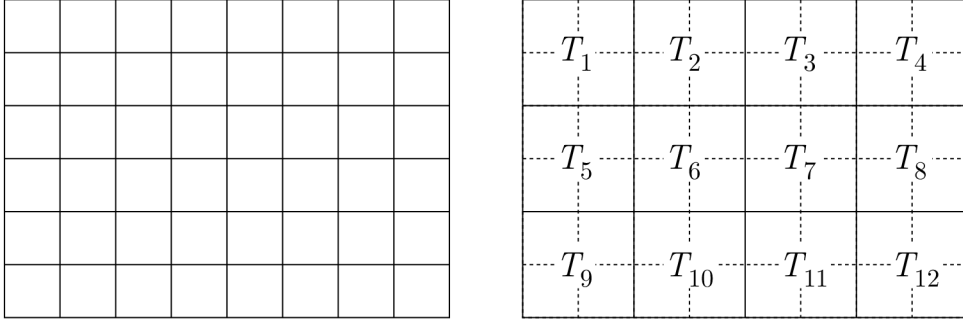
(iii) Vyčíslíme

$$\mu_{\text{O}}^{(t)} = \frac{1}{|M_{\text{O}}^{(t)}|} \sum_{(j,k) \in M_{\text{O}}^{(t)}} i_{in}(j, k), \quad \mu_{\text{B}}^{(t)} = \frac{1}{|M_{\text{B}}^{(t)}|} \sum_{(j,k) \in M_{\text{B}}^{(t)}} i_{in}(j, k).$$

(iv) Položíme $T^{(t+1)} = (\mu_{\text{O}}^{(t)} + \mu_{\text{B}}^{(t)})/2$ a pokračujeme od bodu (ii).

(v) Výpočet zastavíme po dosiahnutí podmienky $|T^{(t)} - T^{(t+1)}| < \varepsilon$ pre zvolené $\varepsilon \in \mathbb{R}$.

Klasické prahovanie môže zlyhať, ak je pozadie obrazu nerovnomerné – optimálna prahová hodnota môže byť pre jeho rôzne časti výrazne odlišná. V takom prípade je nutné prahovať *adaptívne* – obraz rozdeliť a jeho časti prahovať samostatne. Definičný obor M obrazu $i : M \rightarrow R$ delíme na disjunktné, vo väčšine prípadov obdĺžnikové podmnožiny, ktorých zjednotením je celá množina M . Ak to situácia vyžaduje, proces je možné ďalej zdokonaľiť určením vlastnej prahovej hodnoty pre každý pixel obrazu vhodnou metódou interpolácie medzi prahovými hodnotami jeho častí.



Obr. 2.6: Obraz a jeho rozklad na 12 podmnožín s vlastnými prahovými hodnotami T_1, \dots, T_{12} .

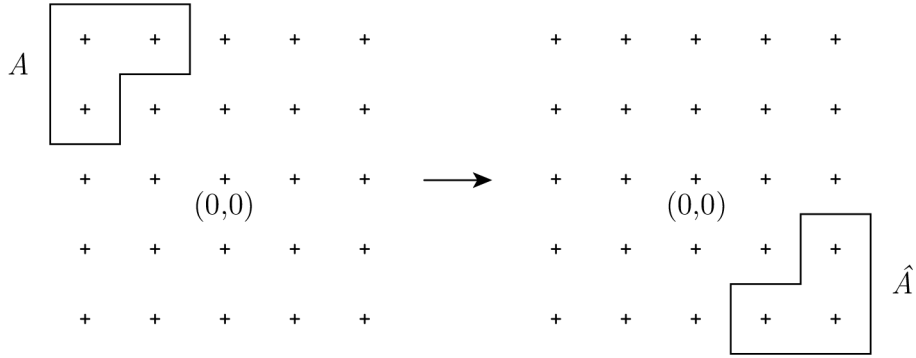
2.5 Morfológické operácie

V súlade s poznámkou 2.1.1 sa môžeme na objekty v binárnom obraze pozerat ako na množiny pixelov. Vychádzajúc zo základných množinových operácií, matematická morfológia poskytuje teoretický aparát umožňujúci skúmať a spracovávať tvar obrazových štruktúr. Základne morfológické operácie sú definované prostredníctvom známych operácií zjednotenia, prieniku a komplementu, a o niečo menej používaných operácií translácie a reflexie.

Definícia 2.5.1 (Reflexia množiny). Nech $A \subset \mathbb{Z}^2$. Množinu

$$\hat{A} = \{b \mid b = -a, \text{ kde } a \in A\}$$

nazveme *reflexiou* množiny A .

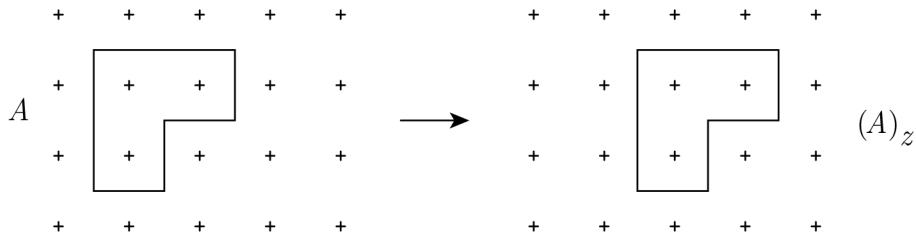


Obr. 2.7: Reflexia \hat{A} množiny A .

Definícia 2.5.2 (Translácia množiny). Množinu

$$(A)_z = \{b \mid b = a + z, \text{ kde } a \in A\}$$

nazveme *transláciou* množiny A bodom $z = (z_1, z_2) \in \mathbb{Z}^2$.



Obr. 2.8: Translácia $(A)_z, z = (0, 1)$ množiny A .

2.5.1 Dilatácia a erózia

Nasledujúce definície predstavujú základné morfológické úpravy binárneho obrazu. Množinu jeho objektových pixelov upravujú prostredníctvom *štruktúrneho elementu*. Medzi základné operácie patria dilatácia a erózia, pomocou ktorých následne zadefinujeme operácie otvorenia a uzavretia.

Definícia 2.5.3 (Dilatácia množiny). Nech $A, B \subseteq \mathbb{Z}^2$. Množinu

$$A \oplus B := \{z \in \mathbb{Z}^2 \mid (\hat{B})_z \cap A \neq \emptyset\}$$

nazveme *dilatáciou* množiny A štruktúrnym elementom B . Ekvivalentne,

$$A \oplus B := \bigcup_{b \in B} (A)_b.$$

Definícia 2.5.4 (Erózia množiny). Nech $A, B \subseteq \mathbb{Z}^2$. Množinu

$$A \ominus B := \{z \in \mathbb{Z}^2 \mid (B)_z \subseteq A\}$$

nazveme *eróziou* množiny A štruktúrnym elementom B . Ekvivalentne,

$$A \ominus B := \bigcap_{b \in B} (A)_{-b}.$$

Poznámka 2.5.1 (Rozmer a značenie štruktúrneho elementu). Štruktúrny element $B \in \mathbb{Z}^2$ nazveme štruktúrnym elementom (rozmeru) $(2n + 1) \times (2n + 1)$, ak

$$n = \max_{s \in S} \{s\}, \quad \text{kde } S = \{s \mid s = \max\{b_1, b_2\}, (b_1, b_2) \in B\} \quad (2.5)$$

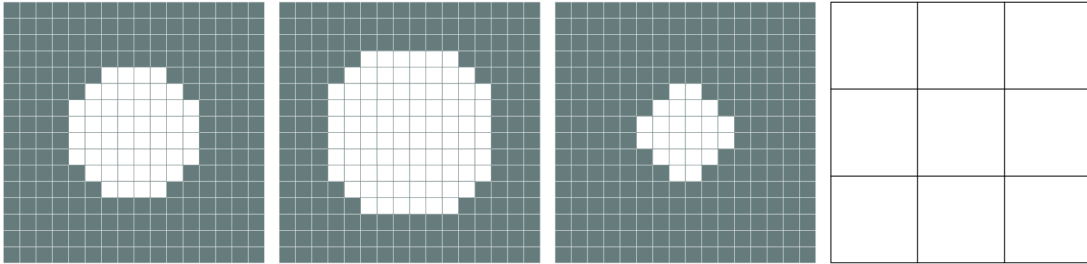
a zapisovať ho budeme v tvare mriežky

$$B = \begin{bmatrix} b_{-n,-n} & \cdots & b_{-n,n} \\ \vdots & \ddots & \vdots \\ b_{n,-n} & \cdots & b_{n,n} \end{bmatrix}, \quad (2.6)$$

pričom na pozíciu $b_{j,k}$ zapíšeme symbol „1“, ak $(j, k) \in B$, a symbol „×“, ak $(j, k) \notin B$.

Za predpokladu, že počiatok $(0, 0) \in B$, výsledkom dilatácie $A \oplus B$ je množina obsahujúca všetky body množiny A a tie body, ktoré patria do množiny A posunutej o $b \in B$. Dilatácia tak množinu objektových pixelov rozširuje. Výsledkom erózie $A \ominus B$ je množina bodov, ktoré patria zároveň do A a do A posunutej o $-b$, $b \in B$. Erózia potom množinu objektových pixelov zužuje. Všeobecne, pixel (j, k) binárneho obrazu i je po erózii vyhodnotený ako objektový práve vtedy, keď

$$i(j + b_1, k + b_2) = 1 \quad \text{pre } \forall (b_1, b_2) \in B. \quad (2.7)$$



Obr. 2.9: Pôvodný binárny obraz, binárny obraz po dilatácii, binárny obraz po erózii a užitý štruktúrny element 2.6. Hranice pixelov sú zvýraznené za účelom prehľadnosti.

Veta 2.5.1 (Vlastnosti dilatácie a erózie). Nech $A, B, B_1, B_2 \in \mathbb{Z}^2$. Potom platí

- (i) $(A \oplus B_1) \oplus B_2 = A \oplus (B_1 \oplus B_2)$,
- (ii) $(A \ominus B_1) \ominus B_2 = A \ominus (B_1 \oplus B_2)$,
- (iii) $(A \ominus B)^c = A^c \ominus \hat{B}$,

Asociatívnosť (i) nám umožňuje implementovať „veľké“ dilatácie ako niekoľko po sebe nasledujúcich „malých“ dilatácií. Podobne, vlasnosť (ii) umožňuje to isté pre eróziu. Predpokladáme pri tom, že štruktúrny element B dokážeme vyjadriť ako

$$B = B_1 \oplus B_2 \oplus \cdots \oplus B_n, \quad n \in \mathbb{N}.$$

Vzťah (iii) vyjadruje *dualitu* týchto operácií – dilatácia objektových pixelov je to isté, ako erózia pozadia s reflektovaným štruktúrnym elementom. Vďaka dualite nie sme pri implementácii nútení uvažovať obe operácie zároveň – eróziu objektov v obraze dokážeme vykonať prostredníctvom dilatácie pozadia a naopak. Navyše je postačujúce, aby sme dokázali tieto operácie vykonávať na množine objektových pixelov a zároveň aby sme dokázali z obrazu vytvoriť jeho komplement. Dilatácia, resp. erózia pozadia obrazu je potom komplementom dilatácie, resp. erózie množiny objektových pixelov komplementu komplementu pôvodného obrazu.

2.5.2 Uzavretie a otvorenie

Dilatácia štruktúrnym elementom B a následná erózia týmto elementom umožňuje z obrazu odstrániť šum v podobe „dier“ v množine objektových pixelov, t. j. nežiadúce pixely *pozadia* spôsobujúce nesúvislosť objektov. Naopak, erózia a následná dilatácia z obrazu odstraňuje šum v podobe nežiadúcich *objektových* pixelov. Tieto operácie sú nazývané *uzavretie a otvorenie*.

Definícia 2.5.5 (Uzavretie množiny). Nech $A, B \subseteq \mathbb{Z}^2$. Množinu

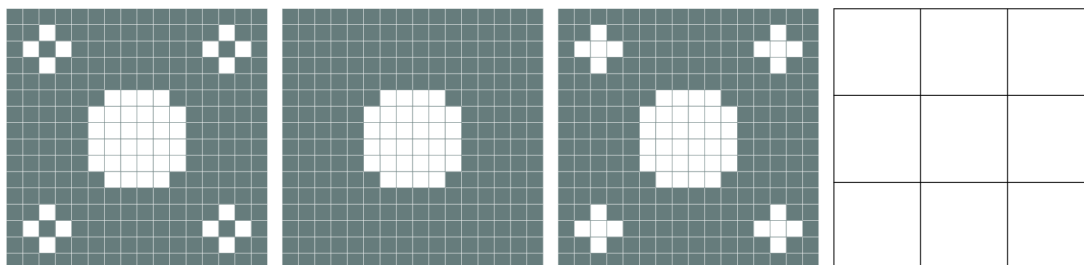
$$A \bullet B := (A \oplus B) \ominus B$$

nazveme *uzavretím* množiny A štruktúrnym elementom B .

Definícia 2.5.6 (Otvorenie množiny). Nech $A, B \subseteq \mathbb{Z}^2$. Množinu

$$A \circ B := (A \ominus B) \oplus B$$

nazveme *otvorením* množiny A štruktúrnym elementom B .



Obr. 2.10: Pôvodný binárny obraz, binárny obraz po otvorení, binárny obraz po uzavretí a užitý štruktúrny element 2.6. Hranice pixelov sú zvýraznené za účelom prehľadnosti.

2.5.3 Hit-or-miss transformácia

Podmienka 2.7 hovorí, ktoré pixely z okolia skúmaného pixelu musia byť *objektové*, aby bol aj skúmaný pixel po erózií vyhodnotený ako objektový. Túto podmienku dokážeme zosilniť tak, že z okolia skúmaného pixelu Navyše vyberieme pixely, od ktorých budeme vyžadovať, aby boli pixelmi *pozadia*. Táto operácia je nazývaná *hit-or-miss* transformácia a je základným prostriedkom na rozpoznávanie vzorov v binárnom obraze.

Definícia 2.5.7 (Hit-or-miss transformácia). Nech $A, B_1, B_2 \subseteq \mathbb{Z}^2$, $B_1 \cap B_2 = \emptyset$. Množinu

$$A \odot B := (A \ominus B_1) \cap (A^c \ominus B_2),$$

kde A^c značí komplement množiny A , nazveme *hit-or-miss transformáciou* množiny A (zloženým) štruktúrnym elementom $B = (B_1, B_2)$.

Zložené štruktúrne elementy budeme značiť v tvare mriežky 2.6, pričom na pozíciu $b_{j,k}$ zapíšeme symbol „1“, ak $(j, k) \in B_1$, symbol „0“, ak $(j, k) \in B_2$ a symbol „×“, ak $(j, k) \notin B_1 \cup B_2$. Pomocou hit-or-miss transformácie teraz dokážeme definovať operáciu stenčenia a sekvenčného stenčenia.

0	0	1	0	0	0	0
0	1	1	0	1	1	0
1	1	0	0	1	1	0
1	1	1	1	1	1	1
1	1	1	1	1	1	1

0	0	0	0	0	0	0
0	1	0	0	1	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0

Obr. 2.11: Hit-or-miss transformácia štruktúrnym elementom B^8 (vzťahy 2.8)

Definícia 2.5.8 (Stenčenie). Nech $A, B_1, B_2 \subseteq \mathbb{Z}^2$, $B_1 \cap B_2 = \emptyset$. Množinu

$$A \otimes B := A \setminus (A \odot B)$$

nazveme stenčením množiny A (zloženým) štruktúrnym elementom $B = (B_1, B_2)$.

Definícia 2.5.9 (Sekvenčné stenčenie). Nech $A \subseteq \mathbb{Z}^2$ a nech $\{B^k\}_{k=1}^n$, kde $n \in \mathbb{N}$, je postupnosť zložených štruktúrnych elementov $B^k = (B_1^k, B_2^k)$, $B_1^k, B_2^k \subseteq \mathbb{Z}^2$. Množinu

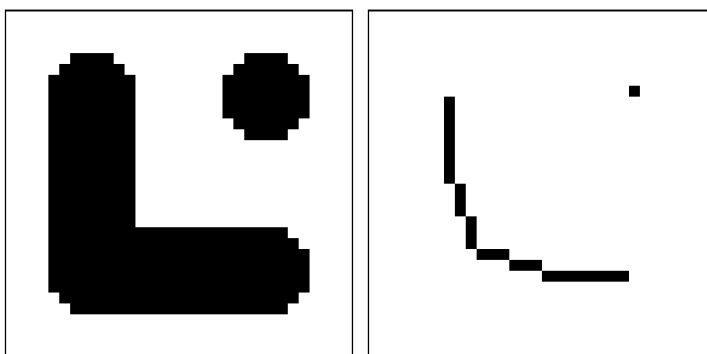
$$A \otimes \{B^k\}_{k=1}^n := ((\dots((A \otimes B^1) \otimes B^2) \dots) \otimes B^n)$$

nazveme *sekvenčným stenčením* množiny A postupnosťou (zložených) štruktúrnych elementov B^1, \dots, B^n .

Pri sekvenčnom stenčení je obvykle používaná nasledujúca postupnosť štruktúrnych elementov pozostávajúca z B^1, B^2 a ich troch rotácií o $90^\circ, 180^\circ$ a 270° okolo počiatku.

$$\begin{aligned}
 B^1 &= \begin{bmatrix} 0 & 0 & 0 \\ \times & 1 & \times \\ 1 & 1 & 1 \end{bmatrix}, & B^2 &= \begin{bmatrix} \times & 0 & 0 \\ 1 & 1 & 0 \\ \times & 1 & \times \end{bmatrix}, & B^3 &= \begin{bmatrix} 1 & \times & 0 \\ 1 & 1 & 0 \\ 1 & \times & 0 \end{bmatrix}, & B^4 &= \begin{bmatrix} \times & 1 & \times \\ 1 & 1 & 0 \\ \times & 0 & 0 \end{bmatrix}, \\
 B^5 &= \begin{bmatrix} 1 & 1 & 1 \\ \times & 1 & \times \\ 0 & 0 & 0 \end{bmatrix}, & B^6 &= \begin{bmatrix} \times & 1 & \times \\ 0 & 1 & 1 \\ 0 & 0 & \times \end{bmatrix}, & B^7 &= \begin{bmatrix} 0 & \times & 1 \\ 0 & 1 & 1 \\ 0 & \times & 1 \end{bmatrix}, & B^8 &= \begin{bmatrix} 0 & 0 & \times \\ 0 & 1 & 1 \\ \times & 1 & \times \end{bmatrix}.
 \end{aligned}
 \tag{2.8}$$

Sekvenčné stenčenie je opakované iteračne až do stavu idempotencie, t. j. do vtedy, kým sa v obraze nenachádza žiaden zo vzorov 2.8. Výsledkom je obraz pozostávajúci z čiar širokých 1 pixel zachovávajúci súvislosť komponent.



Obr. 2.12: Pôvodný obraz a jeho sekvenčné stenčenie

2.5.4 Geodetická dilatácia

Operácia dilatácie bola v sekcii 2.5.1 bola zavedená ako zjednotenie posunutí vstupnej množiny, ktoré sú určené štruktúrnym elementom. Niektoré situácie si môžu vyžadovať výslednú množinu obmedziť tak, aby výsledkom nebolo celé zjednotenie, ale len jeho vhodná podmnožina.

Definícia 2.5.10 (Geodetická dilatácia). Nech $B, X \subseteq \mathbb{Z}^2$. a nech $A \subseteq X$ Množinu

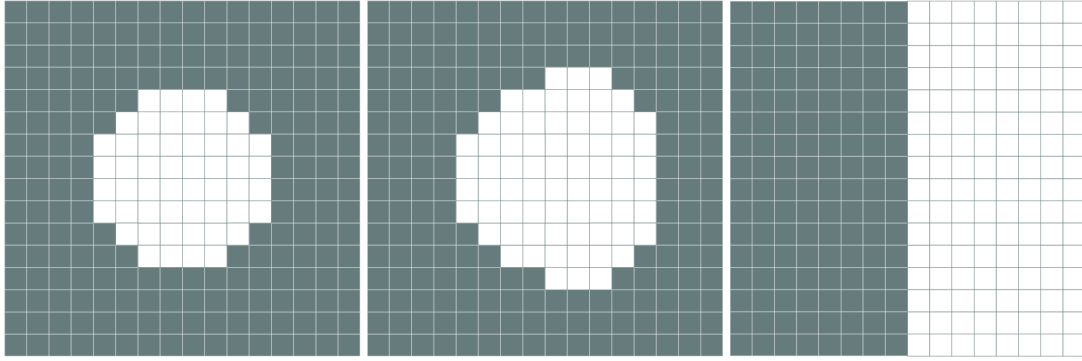
$$A \oplus_X B := (A \oplus B) \cap X$$

nazveme *geodetickou dilatáciou* množiny A štruktúrnym elementom B s maskou X .

Maska X tak pôsobí ako obmedzenie šírenia dilatácie – vymedzuje body, o ktoré môže byť množina A rozšírená. Okrem predpokladu $A \subseteq X$, ktorý má pri aplikácii v obraze podobu

$$i(j, k) \leq i_X(j, k) \quad \text{pre} \quad j = 1, \dots, h, \quad k = 1, \dots, w,$$

ďalej vyžadujeme, aby vstupný obraz i a maskovací obraz i_X mali totožné rozmery, resp. definičný obor.



Obr. 2.13: Pôvodný binárny obraz, binárny obraz po geodetickej dilatácii a použitý maskovací obraz. Hranice pixelov sú zvýraznené za účelom prehľadnosti.

2.6 Flood fill

Definícia 2.6.1 (Flood fill). Nech $i_1 : M \rightarrow \{0, 1\}$ je binárny obraz, $M_O \neq \emptyset$ množina jeho objektových pixelov, $M_B \neq \emptyset$ množina pixelov pozadia, $p \in M_B$ a $n, m \in \mathbb{N}$. Ďalej nech

$$M_B = \bigcup_{k=1}^n K_k,$$

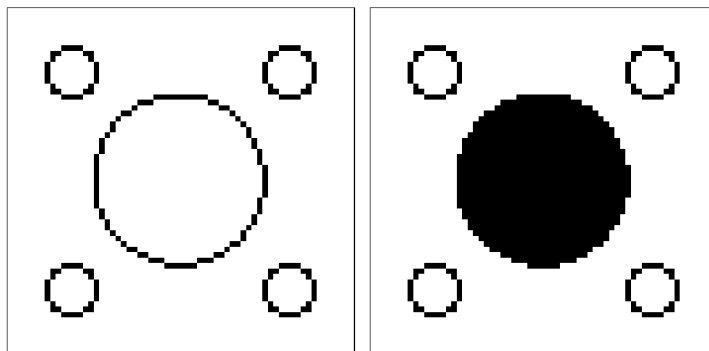
kde K_1, \dots, K_n sú η -komponenty množiny M_B . Operácia *flood fill* je zobrazenie

$$(i_1, p) \mapsto i_2,$$

pričom obraz i_2 je binárny obraz s definičným oborom M splňujúci

$$i_2(q) = \begin{cases} 1, & \text{ak } q \in K, \\ i_1(q) & \text{inak.} \end{cases}$$

kde $K \in \{K_1, \dots, K_n\}$ je tá η -komponenta množiny M_B , ktorá obsahuje p .



Obr. 2.14: Obraz a jeho uprava operáciou flood fill kde $\eta = \eta_4$ a p je jeho stredový pixel.

Operácia flood fill tak upravuje binárny obraz rozšírením jeho množiny objektových pixelov vstupného obrazu o tie body, ktoré patria do jednej zvolenej komponenty jeho množiny pixelov pozadia. Všeobecne býva definovaná pre ľubovoľný (nie nevyhnutne binárny) obraz a pre ľubovoľný počiatočný pixel. Na účely práce je však uvedená definícia postačujúca. Implementácia je realizovateľná rekurzívnym algoritmom [10]

Algoritmus 1: Flood fill (pre $\eta = \eta_8$)

vstupné dáta: $p = (j, k)$

```

1 if  $i(p) = 0$  then
2   Nastav  $i(p) := 1$ 
3   Vykonaj Flood fill s  $p = (j + 1, k)$ 
4   Vykonaj Flood fill s  $p = (j, k + 1)$ 
5   Vykonaj Flood fill s  $p = (j - 1, k)$ 
6   Vykonaj Flood fill s  $p = (j, k - 1)$ 
7   Vykonaj Flood fill s  $p = (j + 1, k + 1)$ 
8   Vykonaj Flood fill s  $p = (j + 1, k - 1)$ 
9   Vykonaj Flood fill s  $p = (j - 1, k + 1)$ 
10  Vykonaj Flood fill s  $p = (j - 1, k - 1)$ 
11 end

```

2.7 Houghova transformácia

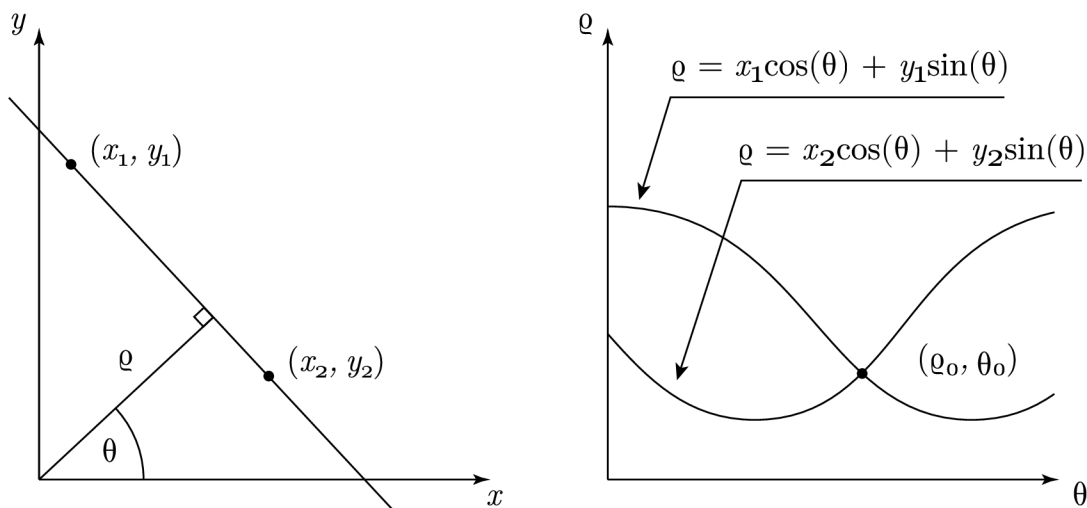
Houghova transformácia je numerická metóda na rozpoznávanie priamok, kružníc a iných matematicky vyjadriteľných kriviek v binárnom obraze. S ohľadom na ciele práce sa v nasledujúcom opise zameriame na priamky

$$y = ax + b.$$

Aby sme však zohľadnili aj priamky kolmé na os x , budeme používať normálový tvar

$$\rho = x \cos \theta + y \sin \theta, \tag{2.9}$$

kde ρ je kolmá vzdialenosť priamky od počiatku a θ je uhol, ktorý zvierá táto kolmica s osou x . Množina všetkých priamok v obraze tvorí dvojparametrickú sústavu. Ak obmedzíme θ na interval $[0, \pi)$, priamka je určená práve jedným bodom (θ_0, ρ_0) v parametrickom priestore. Potom kolinéarne body v rovine (x, y) zodpovedajú sústave sinusiodálnych kriviek s práve jedným priesečníkom (θ_0, ρ_0) , ktorý túto priamku určuje.



Obr. 2.15: Dva body určujúce priamku v rovine (x, y) a odpovedajúce sinusoidy určujúce parametre tejto priamky v parametrickom priestore.

Pri detekcii priamok v obraze i s rozmermi $h \times w$ pristupujeme k numerickému riešeniu. Premenné v rovnici 2.9 transformujeme tak, aby zodpovedali orientácii osí a súradniciam pixelov obrazu

$$j = -y, \quad k = x \quad \implies \quad \varrho = k \cos \theta - j \sin \theta.$$

Rovinu ϱ, θ obmedzíme na obdĺžnik $R \times T$, kde

$$R = [0, \frac{1}{2}\sqrt{w^2 + h^2}], \quad T = [0, \pi)$$

a kvantujeme delením

$$\begin{aligned} 0 = r_0 \leq r_1 \leq r_2 \leq \dots < r_n = \frac{1}{2}\sqrt{w^2 + h^2}, \\ 0 = t_0 \leq t_1 \leq t_2 \leq \dots \leq t_m = \pi, \\ n, m \in \mathbb{N}, \end{aligned}$$

pričom z každého podintervalu zvolíme jednu reprezentatívnu hodnotu

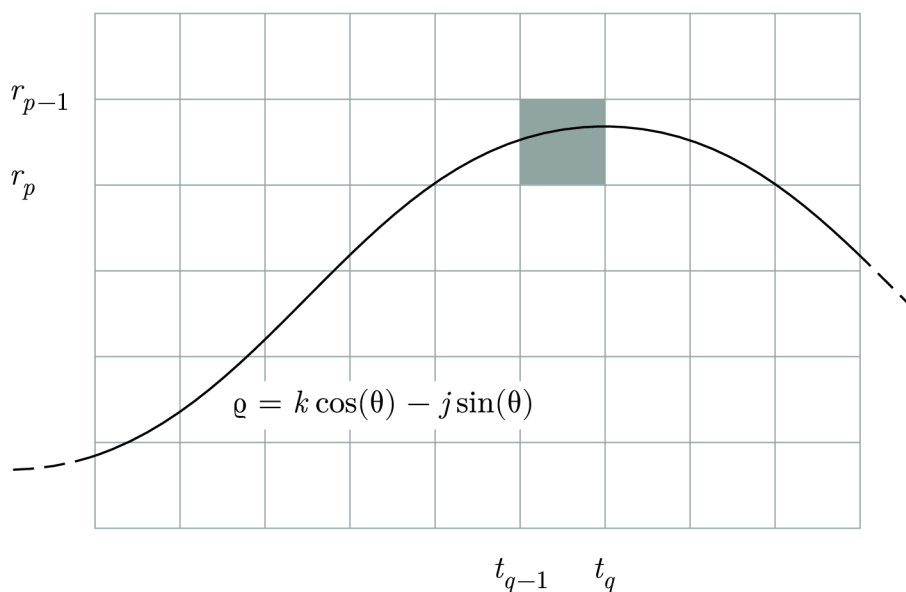
$$\begin{aligned} \varrho_1, \varrho_2, \dots, \varrho_n, \\ \theta_1, \theta_2, \dots, \theta_m. \end{aligned}$$

Ďalej zavedieme $n \times m$ akumuláčnú maticu $\mathbf{A} = [a_{pq}]$, ktorá je na začiatku výpočtu nulová. V každom kroku numerického riešenia potom skúmame objektový pixel (j, k) obrazu a hodnotu a_{pq} matice \mathbf{A} zvýšime o 1, ak

$$\{(\theta, \varrho) \mid \varrho = k \cos \theta - j \sin \theta\} \cap [r_{p-1}, r_p] \times [t_{q-1}, t_q] \neq \emptyset,$$

t. j. ak sinusoida daná skúmaným pixelom prechádza zodpovedajúcou bunkou mriežky získanej delením parametrického priestoru.

Postup opakujeme pre všetky objektové pixely skúmaného obrazu. Reprezentatívne hodnoty ϱ_p, θ_q podintervalov $[r_{p-1}, r_p], [t_{q-1}, t_q]$ prislúchajúce *maximálnym* prvkom a_{pq} matice \mathbf{A} potom aproximujú parametre ϱ a θ priamok nachádzajúcich sa v obraze – kolineárne body v obraze totiž spôsobujú opakovanú inkrementáciu zodpovedajúceho prvku v akumuláčnej matici \mathbf{A} .



Obr. 2.16: Sinusoida v parametrickom priestore a jeho delenie mriežkou.

Výpočetná náročnosť Houghovej transformácie narastá s počtom parametrov potrebných na určenie detekovaného objektu. Pre prípad priamok je však výpočet relatívne nenáročný a implementácia je realizovateľná nasledujúcim algoritmom

Algoritmus 2: Houghova transformácia

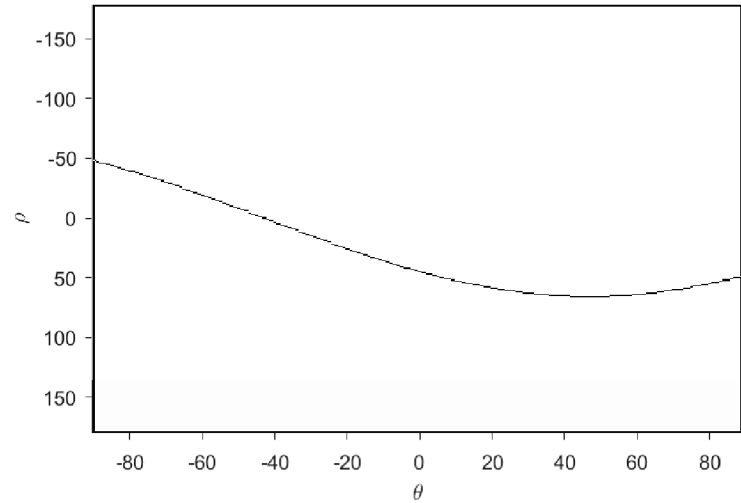
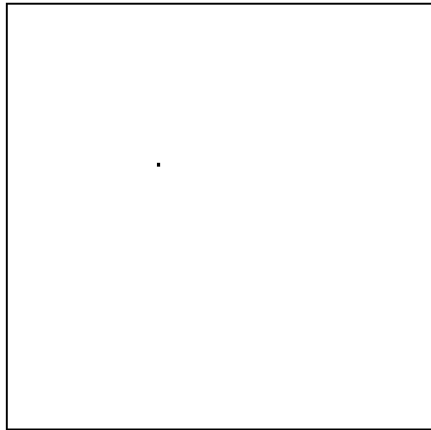
vstup : Binárny obraz i o rozmeroch $h \times w$

výstup: Obraz h o rozmeroch $\sqrt{h^2 + w^2} \times 180$

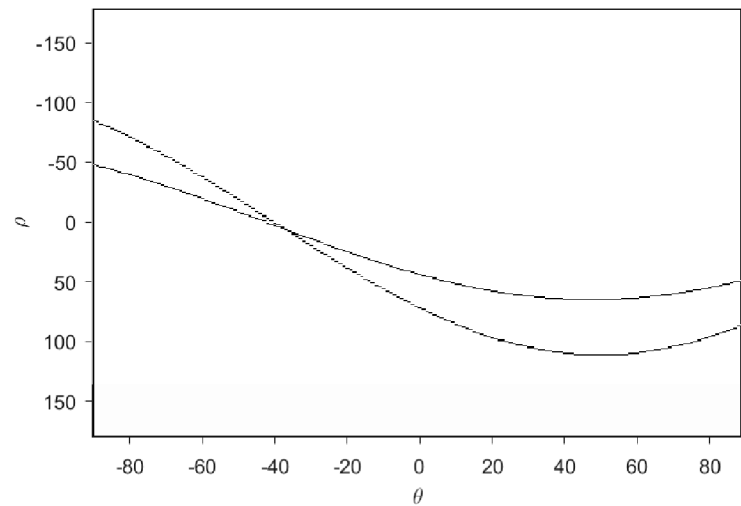
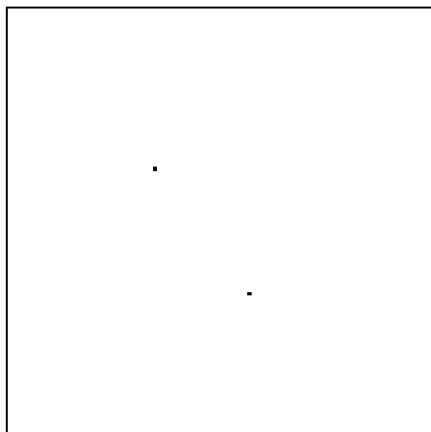
```

1 Inicializuj obraz  $h \equiv 0$ 
2 for each objektový pixel  $p(j, k)$  obrazu  $i$  do
3   for  $\theta := 0$  to 180 do
4      $\rho := k \cos(\theta \cdot \pi/180) - j \sin(\theta \cdot \pi/180)$ 
5      $r := \text{Round}(\rho)$ 
6      $h(r, \theta) := h(r, \theta) + 1$ 
7   end
8 end

```



Obr. 2.17: Obrázok obsahujúci jeden objektový pixel (vľavo) a zodpovedajúca sinusoida v parametrickom priestore (vpravo).



Obr. 2.18: Obrázok obsahujúci dva objektové pixely (vľavo) a zodpovedajúce sinusoidy v parametrickom priestore (vpravo). Priesečník týchto kriviek určuje parametre priamky danej bodmi v pôvodnom obraze.

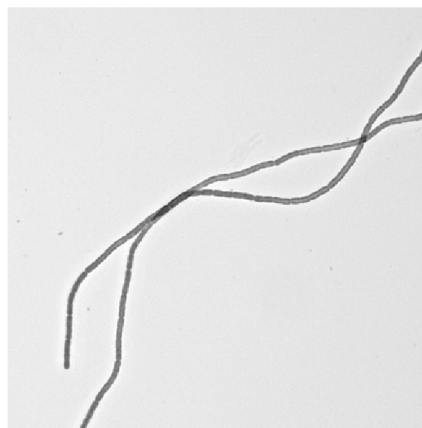
3 Predspracovanie obrazových dát

V nasledujúcej kapitole bude opísaný postup predspracovania obrazových dát – skúmanými obrazmi sú šedotónové snímky z mikroskopu zachytávajúce vlákna morských rias. Tieto vlákna sa všeobecne môžu na rôznych miestach v obraze prekryvať a krížiť. Cieľom je dáta pripraviť tak, aby bolo možné jednotlivé vlákna separovať a následne analyzovať ich vlastnosti. Konkrétne nás zaujíma ich dĺžka – aby sme ju boli schopní odhadnúť, každé vlákno potrebujeme reprezentovať jednou 8-konektívnou krivkou hrúbky 1 pixel.

Prezentovaný postup predspracovania sa opiera o poznatky z publikácie [15] a za cieľ si pokladá surové dáta na požadovaný tvar previesť použitím metód uvedených v kapitole 2. Demonštrovaný bude na vybraných snímkach – prehľad všetkých spracovávaných snímok sa nachádza v prílohe A.



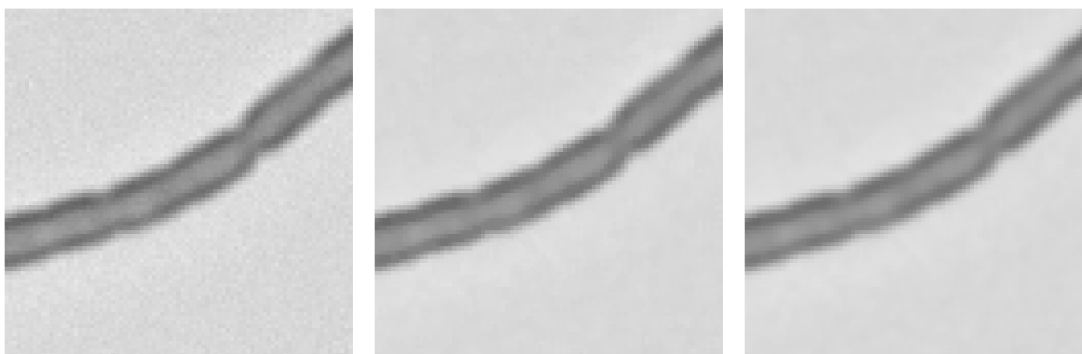
Obr. 3.1: Snímka 1



Obr. 3.2: Snímka 1 (detail)

3.1 Filtrácia

Spracovávané snímky pochádzajú z digitálnej kamery, a sú tak do určitej miery ovplyvnené šumom. Tento šum spôsobuje zrnitosť, ktorá sa môže pri prahovaní a následných morfológických úpravách patologicky prejaviť v podobe malých novovzniknutých segmentov vlákien nevyskytujúcich sa v pôvodnom obraze.



Obr. 3.3: Snímka 1 (detail) pred filtráciou, po aplikácii mediánového filtra a po aplikácii filtra klzavého priemeru.

Na potlačenie impulzného šumu je použitý 3×3 mediánový filter nasledovaný 3×3 lineárnym filtrom klzavého priemeru

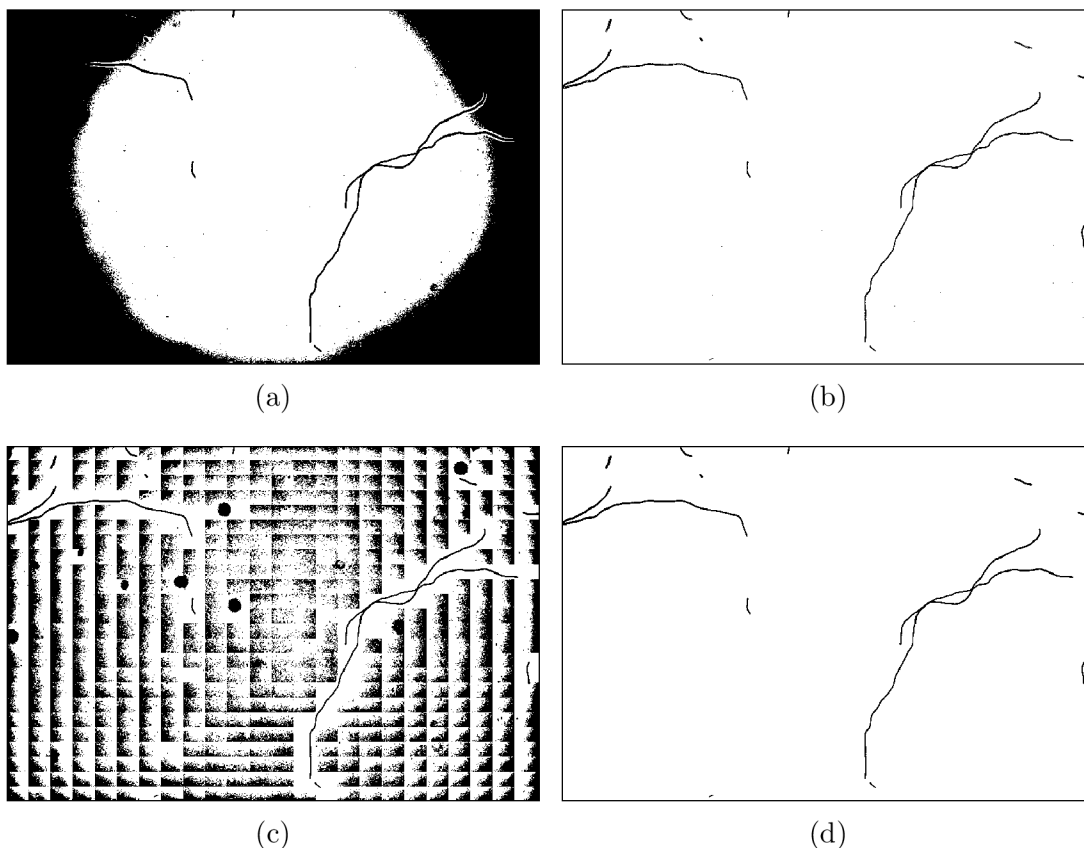
$$\mathbf{G} = \frac{1}{9} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix},$$

ktorý v obraze ďalej potlačuje aditívny šum.

3.2 Segmentácia

Pozorované objekty pozostávajú z pixelov, ktoré majú v porovnaní s pozadím značne nižšie jasové hodnoty. Snímky sú preto dobre segmentovateľné prahovaním.

Vo väčšine prípadov je možné prahovú hodnotu spoľahlivo stanoviť vizuálnym posúdením histogramu. Kvôli nerovnomernosti osvetlenia v snímke spôsobenej najmä vinetáciou však uprednostňujeme adaptívny prístup. Adaptívne prahovanie nám navyše dovoľuje proces automatizovať uplatnením iteračnej metódy stanovenia optimálneho prahu uvedenej v časti 2.4, ktorá by pri klasickom prahovaní neposkytovala vhodný výsledok (obr 3.4a). Obrazy sú tak rozdelené pravidelnou mriežkou na časti prahované samostatne.



Obr. 3.4: (a) prahovanie globálnym prahom získaným iteračným výpočtom, (b) manuálnou voľbou prahu, (c) adaptívne prahovanie s iteračným výpočtom prahu a (d) adaptívne prahovanie s iteračným výpočtom prahu a odstránením oblastí s nepostačujúcim rozptylom (snímka 1).

Nakoľko je prahová hodnota vypočítavaná lokálne a pre každú časť obrazu samostatne, dochádza k nežiadúcej segmentácii oblastí, v ktorých sa pozorované objekty nenachádzajú (obrázok 3.4c). Tento jav je napravený stanovením podmienky minimálneho rozptylu jasových hodnôt. Pri použití mriežky $n \times m$ deliacej definičný obor obrazu

$$i : \{1, \dots, h\} \times \{1, \dots, w\} \rightarrow \{0, \dots, 255\}$$

na podmnožiny

$$\begin{aligned} D_{11} &= \{1, \dots, h_1\} \times \{1, \dots, w_1\}, \\ D_{21} &= \{h_1 + 1, \dots, h_2\} \times \{1, \dots, w_1\}, \\ &\vdots \\ D_{nm} &= \{h_{n-1} + 1, \dots, h_n\} \times \{w_{m-1} + 1, \dots, w_m\}, \end{aligned}$$

kde $0 = h_0 < h_1 < h_2 < \dots < h_n = h$ a $0 = w_0 < w_1 < w_2 < \dots < w_m = w$, sú vypočítané hodnoty rozptylu

$$\text{var } D_{nm} = \frac{1}{|D_{nm}|} \sum_{p \in D_{nm}} (i(p) - \mu_{nm})^2, \quad \text{kde } \mu_{nm} = \frac{1}{|D_{nm}|} \sum_{p \in D_{nm}} i(p),$$

a hodnota priemerného rozptylu

$$\sigma_{\min}^2 := \frac{1}{mn} \sum_{j=1}^n \sum_{k=1}^m D_{nm}.$$

Pixel $p \in D_{nm}$ obrazu i je následne vyhodnotený ako objektový vtedy, ak vyhovuje predpisu 2.4 a oblasť D_{nm} zároveň spĺňa podmienku

$$\text{var } D_{nm} > \epsilon \sigma_{\min}^2,$$

kde $\epsilon > 0$ je vhodne zvolená reálna konštanta citlivosti.

3.3 Morfológická rekonštrukcia

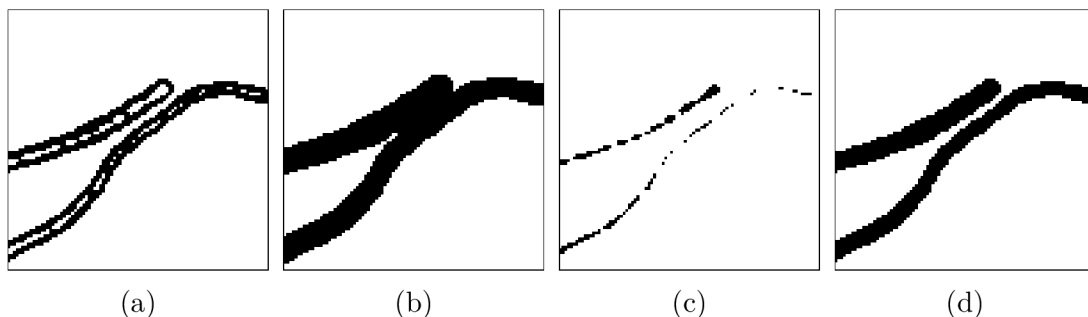
Prahovaním bol získaný binárny obraz rozlišujúci objekty – vlákna – od pozadia. Nakoľko hranica vlákien má na snímkach pozorovateľne nižšiu jasovú hodnotu ako ich vnútro, pri prahovaní dochádza k chybnému vyhodnoteniu vnútra vlákien ako súčasti pozadia na miestach, kde je tento rozdiel dostatočne veľký (obrázok 3.5a).

Vzniknuté diery v objektoch je možné vyplniť opakovanou dilatáciou a eróziou. Tento postup je však aplikovateľný len v prípade, že sa na snímke nenachádzajú miesta, v ktorých sú dve vlákna blízko seba – dilatácia spôsobí ich umelé spojenie (obrázok 3.5b). Aby sme javu predišli, použitá je geodetická dilatácia opísaná v časti 2.5.4 (obrázok 3.5d). Maskovací obraz i_X (obr 3.5c) je zostrojený nasledujúcim spôsobom:

- (i) Z obrazu získaného prahovním sú odstránené komponenty množiny objektových pixelov, ktoré obsahujú okrajové pixely¹.

¹Pixel $p = (j, k)$, ktorý má aspoň jednu súradnicu rovnú jednej, alebo maximálnu.

- (ii) Zvolený je ľubovoľný okrajový pixel $p = (j, k)$.
- (iii) Vykonaná je operácia *flood fill* s počiatočným pixelom p .
- (iv) Z obrazu z predchádzajúceho kroku je vytvorený obraz komplementárny.



Obr. 3.5: (a) detail snímky 2 po prahovaní, (b) spojenie vlákien po dilatácii, (c) maskovací obraz, (d) geodetická dilatácia.

Takto získaný maskovací obraz je binárnym obrazom, v ktorom objektové pixely tvoria tie komponenty množiny pixelov pozadia vstupného obrazu, ktoré neobsahujú jeho okrajové body. Indikuje tak práve diery, ktoré majú byť vyplnené, a prípadné komponenty spôsobené krížením vlákien. Tieto komponenty sú odstraniteľné opakovaním kroku (ii) s vhodnou voľbou p . V skúmaných snímkach to však nebolo nutné. Pre výstupný obraz i_2 tak platí

$$i_2(p) = 1 \iff i_{\oplus}(p) = 1 \wedge i_X(p) = 1,$$

kde i_{\oplus} je obraz získaný klasickou dilatáciou vstupného obrazu i_1 , t. j. množina objektových pixelov M_O^2 výstupného obrazu i_2 je daná

$$M_O^2 = (M_O^1 \oplus_X B) = (M_O^1 \oplus B) \cap M_O^X,$$

kde B je štruktúrny element

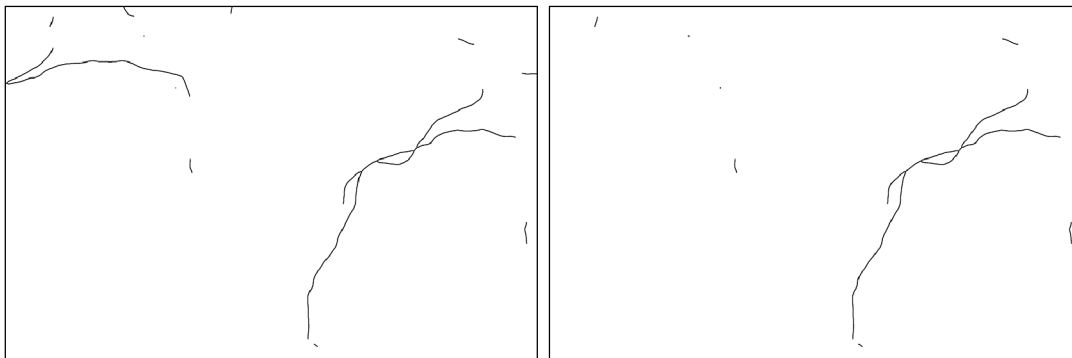
$$B = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} \quad (3.1)$$

Geodetická dilatácia je opakovaná do stavu idempotencie, t. j. pokým ďalšie jej opakovanie už obraz nezmení.

3.4 Stenčovanie a čistenie

Morfologicky rekonštruovaný obraz je následne stenčovaný postupom opísaným v časti 2.5.3. Stenčovanie je uprednostnené pred eróziou, nakoľko vyžadujeme zachovanie súvislosti komponent obrazu. Výsledkom je *skelet* objektov v obraze. S výnimkou miest, kde dochádza k pretínaniu sa vlákien, majú všetky objektové pixely obrazu nanajvýš dva objektové 8-susedné pixely.

V obraze sa však stále nachádzajú nežiadúce objekty. Vlákna, ktoré nie sú na snímke zachytené celé, nie je možné z hľadiska ich dĺžky jednoduchým spôsobom analyzovať. Za predpokladu, že tieto neúplne segmenty predstavujú menej ako 5% obrazových dát, ich je možné z obrazu odstrániť bez toho, aby bola znehodnotená štatistická interpretácia získaných výsledkov. V opačnom prípade je ich nutné pri štatistickom spracovaní považovať za cenzorované dáta. Nakoľko však predmetom práce nie je štatistická analýza, ale separácia pretínajúcich sa kriviek, neúplné vlákna sú z obrazu vždy odstránené.



Obr. 3.6: Obraz po stenčení (vľavo) a po odstránení okrajových objektov (vpravo)

Z obrazu sú následne odstránené aj osamotené pixely, t. j. objektové pixely, ktoré nemajú ani jeden 8-susedný pixel, ktorý by bol tiež objektový. Odstránenie je vykonané operáciou uzavretia so štruktúrnym elementom 3.1.

4 Analýza pretínajúcich sa objektov

V nasledujúcom texte predstavíme algoritmus, ktorý realizuje separovanie kriviek v obraze získanom postupom z predchádzajúcej kapitoly. Začínajúc v jednom z ich koncových bodov v obraze postupujeme pixel po pixely, zaznamenávajúc každý bod krivky do nového výstupného obrazu. V miestach, kde sa krivky krížia, spájajú alebo rozdeľujú, je nutné vykonať rozhodnutie o ich návaznosti. Voľba nasledujúceho pixelu trasovanej krivky je primárne založená na porovnaní sklonu θ priamok, ktoré ju lokálne aproximujú pred a za miestom kríženia. V miestach, kde sa jedna krivka rozdeľuje na dve je navyše využitá informácia o jej minulom chovaní – zohľadnený je predchádzajúci sklon θ_{old} a predchádzajúci smer d_{old} . Nakoniec bude opísaná metóda, ktorou budeme dĺžku separovaných kriviek odhadovať.

4.1 Trasovanie

Vstupný binárny obraz získaný procesom predspracovania opísanom v predchádzajúcej kapitole označme

$$i : M \rightarrow \{0, 1\} \quad (4.1)$$

a ďalej zavedme:

(i) obraz koncových bodov $e : M \rightarrow \{0, 1\}$ daný

$$e(p) = \begin{cases} 1 & \text{ak } i(p) = 1 \wedge |\mathcal{O}_i(p)| = 1, \\ 0 & \text{inak,} \end{cases}$$

(ii) obraz priesečníkov $x : M \rightarrow \{0, 1\}$ daný

$$x(p) = \begin{cases} 1 & \text{ak } i(p) = 1 \wedge |\mathcal{O}_i(p)| > 2, \\ 0 & \text{inak,} \end{cases}$$

(iii) obraz oblúkov $a : M \rightarrow \{0, 1\}$ daný

$$a(p) = i(p) - x(p),$$

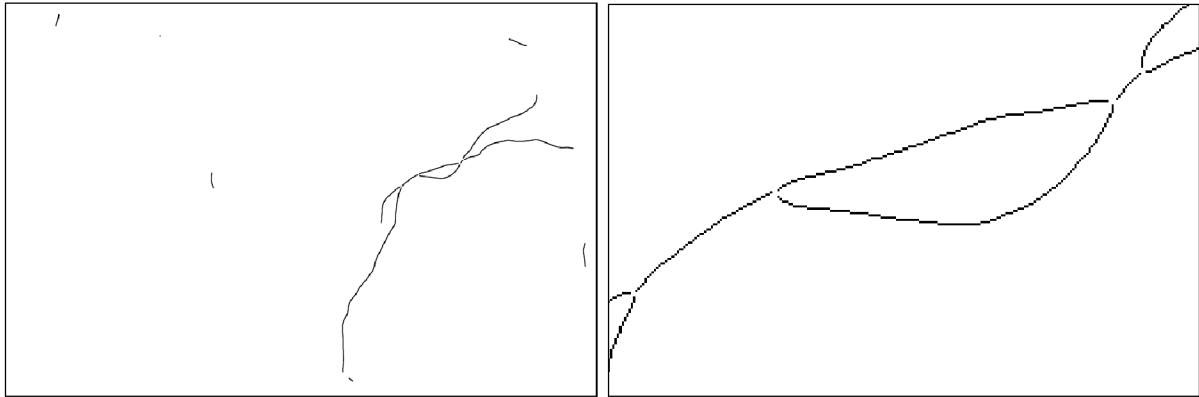
(iv) rozšírený obraz koncových bodov $e^* : M \rightarrow \{0, 1\}$ daný

$$e^*(p) = \begin{cases} 1 & \text{ak } a(p) = 1 \wedge |\mathcal{O}_a(p)| = 1, \\ 0 & \text{inak,} \end{cases}$$

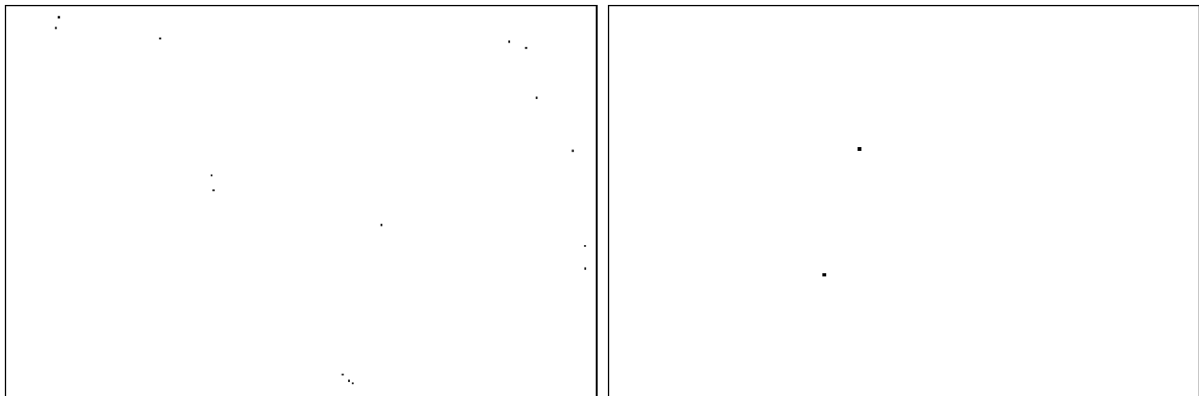
(v) obraz bodov spojenia $s : M \rightarrow \{0, 1\}$ daný

$$s(p) = e^*(p) - e(p),$$

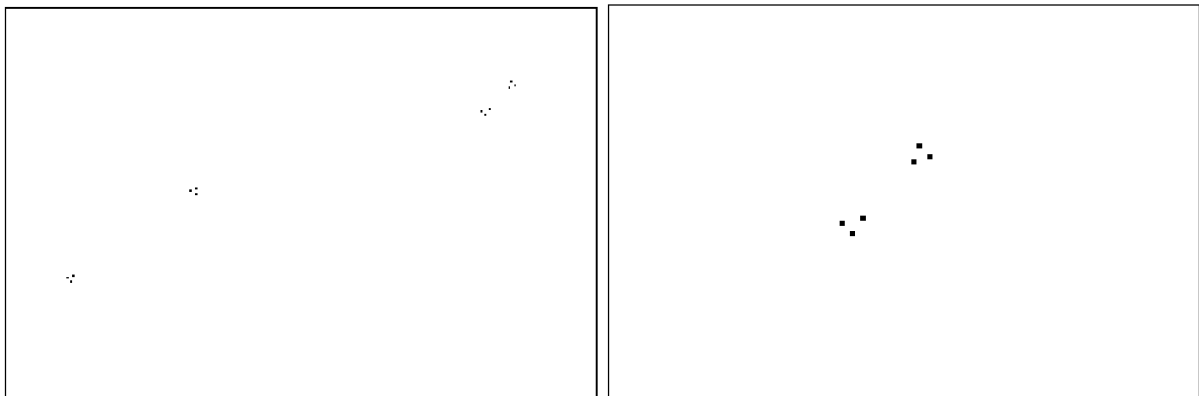
kde $\mathcal{O}_i(p)$ je množina 8-susedných objektových pixelov pixelu p obrazu i zavedená v časti 2.1. Obrazy (i), (iii) a (v) zostrojené z predspracovanej snímky 1 sú zobrazené na obrázkoch 4.1, 4.2 a 4.3.



Obr. 4.1: Obraz oblúkov a (vľavo) a jeho detail (vpravo).



Obr. 4.2: Obraz koncových bodov e (vľavo) a jeho detail (vpravo).



Obr. 4.3: Obraz bodov spojenia s (vľavo) a jeho detail (vpravo).

Poznamenajme, že vyššie uvedené obrazy sú korektne definované. Rozdiel v predpise obrazu oblúkov a (iii) je nulový alebo rovný jednej, nakoľko ak $x(p) = 1$, potom aj $i(p) = 1$ ako priamy následok predpisu obrazu x . Analogicky argument platí pre obraz bodov spojenia s . Nakoniec označme E, X, A, S množiny objektových pixelov obrazov e, x, a, s .

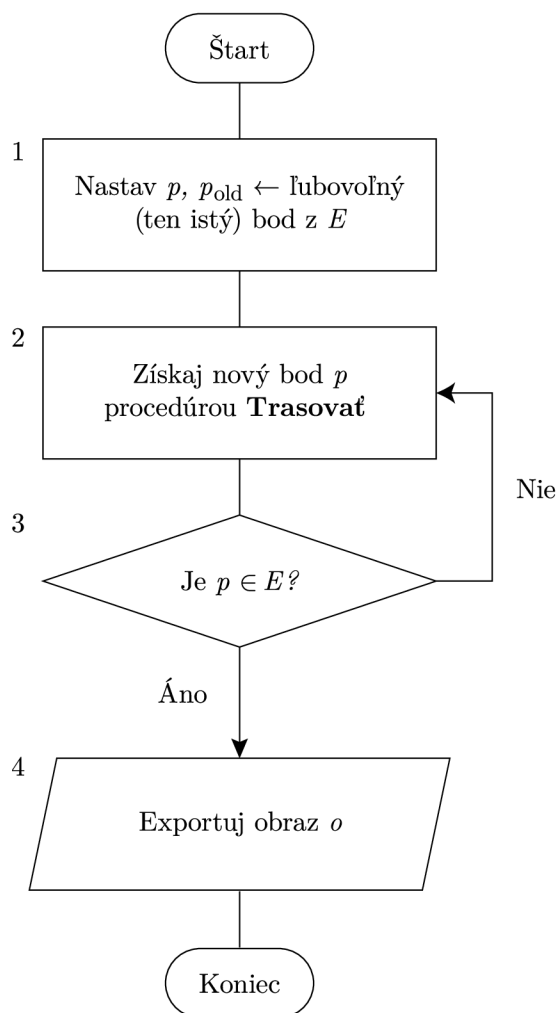
Algoritmus trasovania je cyklický a pozostáva z niekoľkých procedúr. Vstupom je binárny obraz 4.1. Vytvorený je výstupný binárny obraz $o : M \rightarrow \{0, 1\}$ daný

$$o(p) = 0 \quad \text{pre všetky } p \in M$$

a zvolený je počiatočný bod $p \in E$. Následne je volaná procedúra **Trasovať** (obr 4.10), ktorá „posúva“ bod p po aktuálne trasovanej krivke v obraze a . Po každom novom zvolení bodu p je tento bod pridaný do množiny objektových pixelov výstupného obrazu o , t. j.

$$o(p) \leftarrow 1.$$

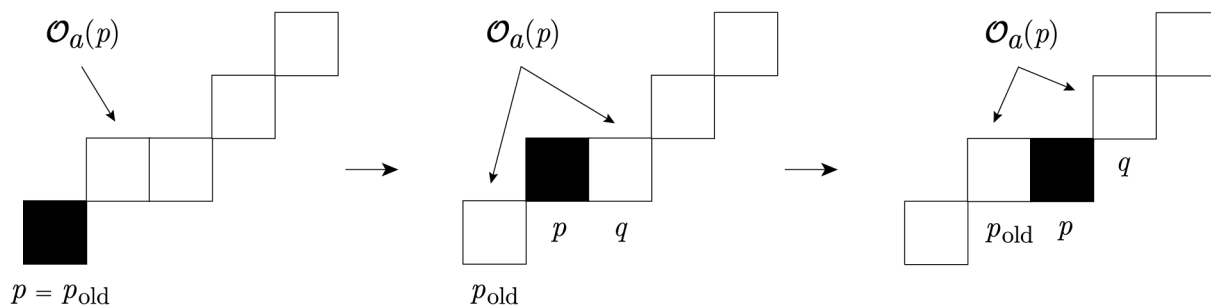
Procedúra **Trasovať** je volaná opakovane do vtedy, kým bod p nie je koncovým bodom krivky. Tento proces je opakovaný pre každý počiatočný bod z množiny E . Výstupom je potom sekvencia obrazov o , z ktorých každý obsahuje práve jednu digitálnu krivku reprezentujúcu jedno z vlákien rias z pôvodnej snímky.



Samotná procedúra **Trasovať** (obrázok 4.10) pozostáva z dvoch častí. V prvej časti (obrázok 4.10, kroky 6–9) sú trasované krivky v obraze oblúkov a – na základe aktuálneho pixelu p je volený pixel nasledujúci, ktorý je vzhľadom k nemu vždy 8-susedný. Nakoľko pre množinu A objektových pixelov obrazu a platí

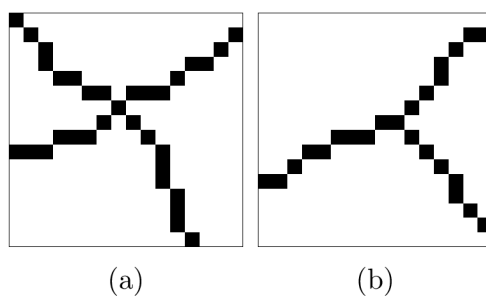
$$1 \leq |\mathcal{O}_a(p)| \leq 2 \quad \text{pre všetky } p \in A,$$

nasledujúci pixel je vždy určený jednoznačne ako také $q \in \mathcal{O}_a(p)$, ktoré nebolo vyhodnocované v minulom kroku (obrázok 4.4). V druhej časti (obrázok 4.10, kroky 12–14) je volaná procedúra **Rozhodni**, ktorá rieši rozhodovanie v miestach spájania, rozchádzania sa alebo kríženia sa vlákien. Tieto miesta sú detekované ako body, ktoré majú iba jeden 8-susedný pixel objektový v obraze a a zároveň nie sú objektové v obraze koncových bodov e .



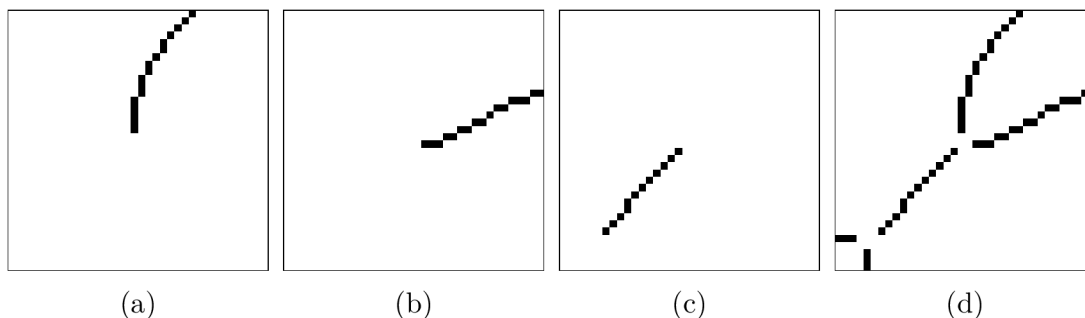
Obr. 4.4: Výber nasledujúceho pixelu v krokoch 6–9 procedúry **Trasovať**

Križenie vlákien v snímke sa v predspracovanom binárnom obraze i prejavuje dvomi spôsobmi – dochádza k dvom rôznym vzorom, ktoré budeme nazývať x -priesečník a y -priesečník. V prvom prípade je miesto križenia charakterizované lokálnou prítomnosťou 4 bodov spojenia, zatiaľ čo v druhom prípade sú lokálne prítomné len 3. Primárne, x -priesečníky vznikajú v miestach, kde sa prekrývajúce vlákna rozchádzajú. Okrem toho však následkom stenčovania v časti 3.4 degenerujú na x -priesečníky aj miesta, kde sa vlákna pretínajú pod dostatočne ostrým uhlom.



Obr. 4.5: (a) x -priesečník, (b) y -priesečník

Procedúra **Rozhodni** má dve varianty. V prípade, že nepoznáme informáciu o predchádzajúcom správaní sa krivky alebo pri prechode algoritmu x -priesečníkom je nasledujúci pixel volený na základe sklonu θ priamky, ktorá aproximuje segment vhodnej dĺžky krivky do priesečníku vstupujúcej. Tento sklon je porovnávaný so sklonom odpovedajúcim segmentom výstupným a preferovaný je ten segment, ktorý minimalizuje vhodne definovanú vzdialenosť sklonov.

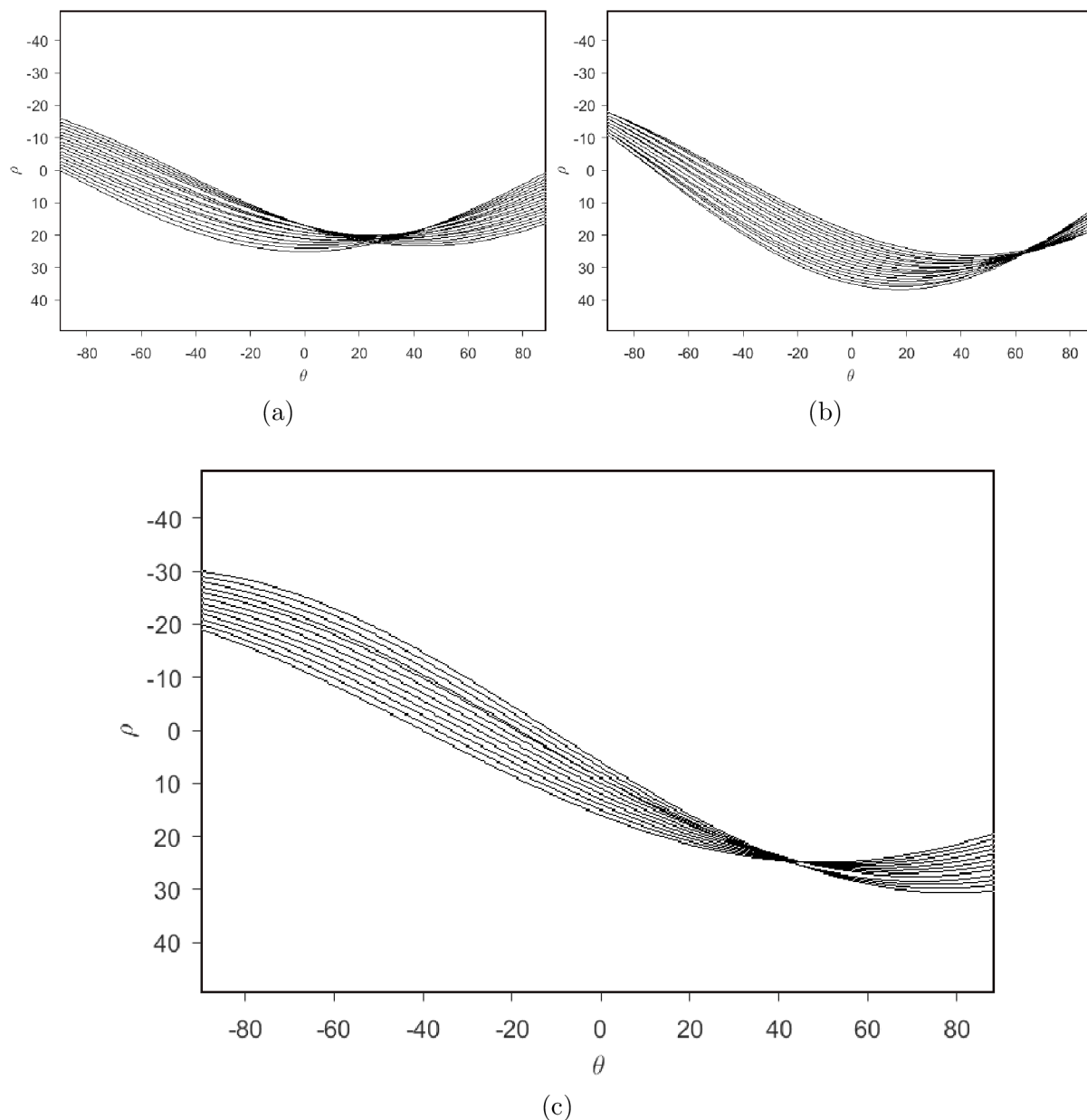


Obr. 4.6: (d) detail obrazu oblúkov a snímky 1, (a) vstupný segment, (b) prvý výstupný segment, (c) druhý výstupný segment

Vyššie spomenutý sklon je získaný Houghovou transformáciou opísanou v časti 2.7. V obrazoch 4.6a–4.6c sú detekované priamky

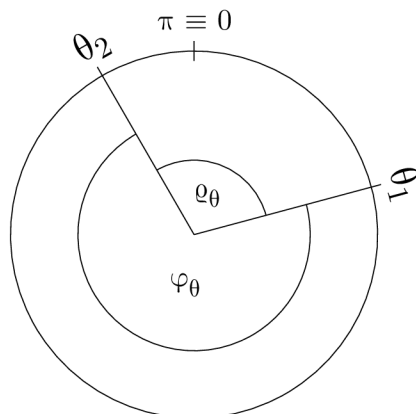
$$\rho = x \cos \theta + y \sin \theta, \quad (4.2)$$

ktorých parametre sú aproximované priesečníkom v sinusoidálnych krivkách v parametrickom priestore na obrázkoch 4.7a–4.7c.



Obr. 4.7: Výsledok Houghovej transformácie (a) segmentu 4.6a, (b) segmentu 4.6b, (c) segmentu 4.6c. Počet priesečníkov môže byť všeobecne väčší ako 1. V tom prípade je hodnota parametru θ odčítaná ako hodnota ich geometrického stredu, t. j. aritmetického priemeru ich súradníc.

Tieto priamky považujeme za lokálnu aproximáciu odpovedajúcich segmentov krivky. Z Houghovej transformácie však nemáme informáciu o smere a veľkosti uhla θ nadobúda len hodnoty z intervalu dĺžky π . Pri porovnávaní veľkosti uhlov je tak nutný modulárny prístup. Uvažovať budeme $\theta \in [0, \pi)$.



Obr. 4.8: Rozdiel sklonu priamok

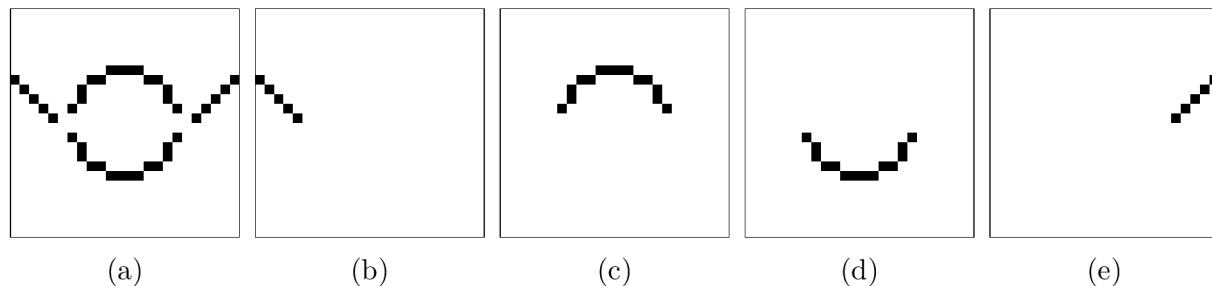
Označme θ_1 sklon odpovedajúci vstupnému a θ_2, θ_3 výstupným segmentom krivky. Ako je vidieť na obrázku 4.8, kde $\theta_2 > \theta_1$, požadovaná hodnota vzdialenosti nemôže byť φ_θ , ale ϱ_θ . Vypočítané sú preto hodnoty $\varrho_\theta(\theta_1, \theta_2)$, $\varrho_\theta(\theta_1, \theta_3)$, kde

$$\varrho_\theta(a, b) := \min\{|b - a|, \pi - |b - a|\}. \quad (4.3)$$

Poznamenajme, že takto definované ϱ_θ je metrikou. Následne je zvolený nasledujúci pixel p tak, aby p patrilo k -temu výstupnému segmentu krivky, ktorý je zvolený tak, aby pre odpovedajúci sklon θ_k platilo

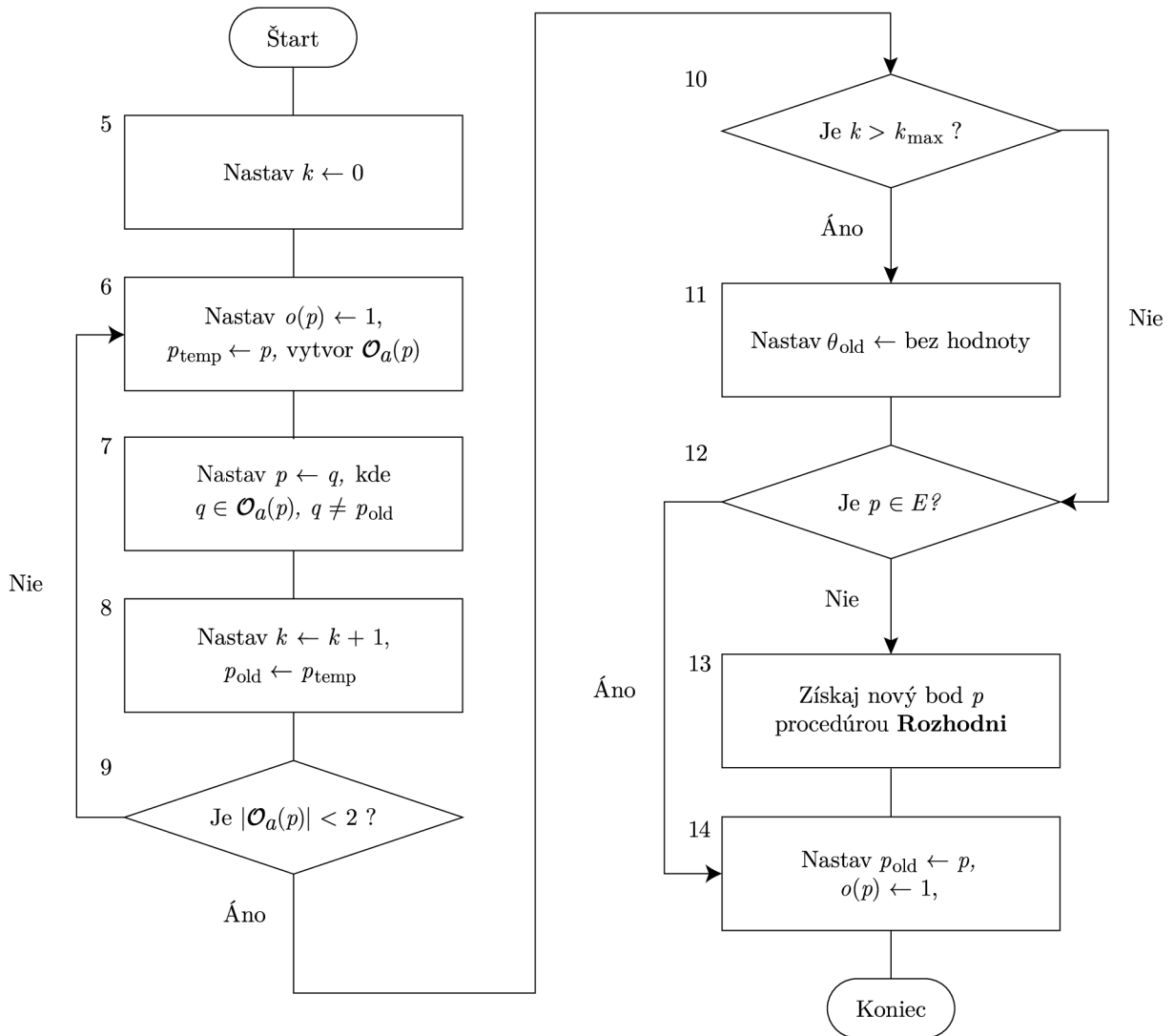
$$\theta_k = \operatorname{argmin}_k \varrho(\theta_1, \theta_k), \quad \text{kde } k \geq 2. \quad (4.4)$$

Po prechode priesečníkom je hodnota θ_1 je uložená do premennej θ_{old} . Táto hodnota je zohľadňovaná druhou variantou procedúri **Rozhodni**, ktorá je volaná vtedy, ak algoritmus vyhodnocuje y-priesečníky (a θ_{old} existuje). Postup je analogický ako v prípade varianty prvej s tým rozdielom, že za vstupný sklon je považovaná práve hodnota θ_{old} .



Obr. 4.9: (a) vzor spôsobujúci zacyklenie, (b), (c), (d), (e) segmenty krivky

V určitých prípadoch môže trasovaná krivka vytvárať vzory, ktoré vedú na zacyklenie algoritmu. K tomuto javu dochádza napríklad v prípade vyzobrazenom na obrázku 4.9a. Pri trasovaní začínajúc ľavým okrajovým pixelom obrazu je po prechode prvým y-priesečníkom uložená hodnota θ_{old} (segment na obrázku 4.9b), ktorá je však takmer totožná so sklonom pravej polovice segmentu na obrázku 4.9d a pri prechode druhým y-priesečníkom teda vyhovuje vzťahu 4.4.



Obr. 4.10: Procedúra **Trasovať**

Aby sme zacykleniu predišli, stanovená je podmienka *dominatného smeru*. Vyhodnocované segmenty sú krivky

$$p_1, \dots, p_n, \quad \text{kde } p_k = (p_k^1, p_k^2), \quad \text{pre } k = 1, \dots, n. \quad (4.5)$$

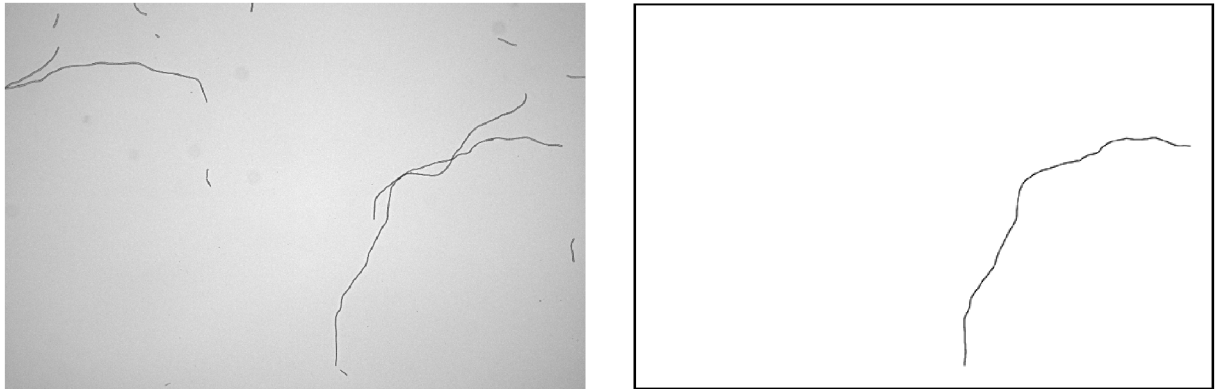
Nakoľko však postupnosti p_1, \dots, p_n a p_n, \dots, p_1 , definujú tú istú krivku, bod p_1 je kvôli jednoznačnosti vždy volený ako prvý trasovaný pixel daného segmentu. Zostrožený je smerový vektor

$$v = p_n - p_1 = (p_n^1 - p_1^1, p_n^2 - p_1^2) = (v^1, v^2). \quad (4.6)$$

pre každý z uvažovaných segmentov a dominantný smer

$$d := \operatorname{argmax}_{k \in \{1,2\}} v^k. \quad (4.7)$$

pre segment vstupujúci. Pri prechode priesečníkom potom vyžadujeme, aby d -ta zložka vstupujúceho a vystupujúceho smerového vektora v mali rovnaké znamienko. Tento postup zacykleniu zabránil vo všetkých spracovávaných snímkach.



Obr. 4.11: Snímka A (vľavo) a výsledná separovaná krivka vo výstupnom obraze o (vpravo)

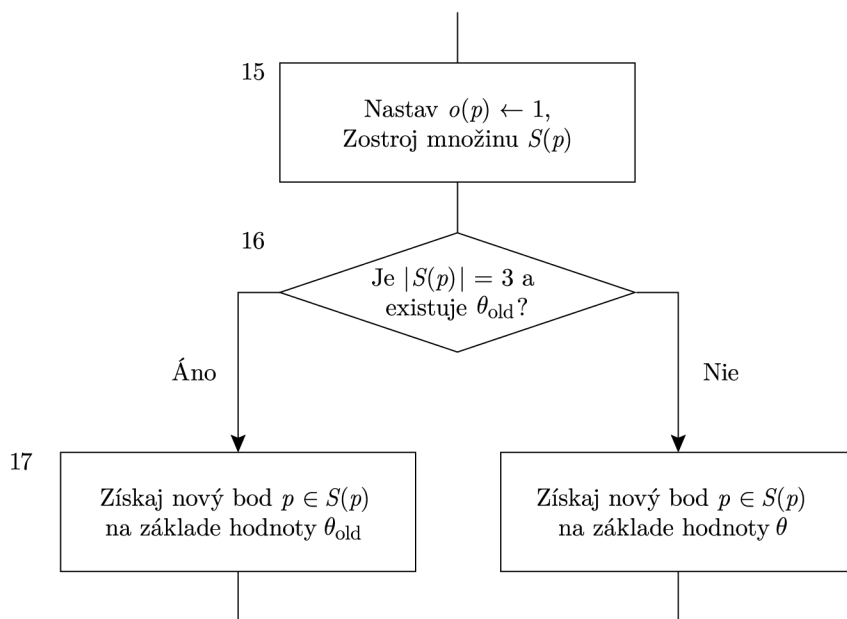
V kroku 17 algoritmu **Rozhodni** (obrázok 4.12) je volený nasledujúci bod p z množiny $S(p)$, ktorú definujeme

$$S(p) := \{q \in S : q \in O_\epsilon(p)\} \quad (4.8)$$

pre vhodné zvolené $\epsilon \in \mathbb{R}^+$. Množina $S(p)$ je teda tvorená bodmi spojenia, ktoré sa nachádzajú v okolí pixelu p . Okrem vyššie uvedených kritérií, na základe ktorých je vykonávané rozhodovanie v miestach priesečníkov, je Navyše zohľadnená informácia o tom, či segment, po ktorom sa má algoritmus vydať, už patrí niektorej z kriviek separovaných v jeho predchádzajúcich cykloch, t. j. či pre niektorý z výstupných obrazov o platí

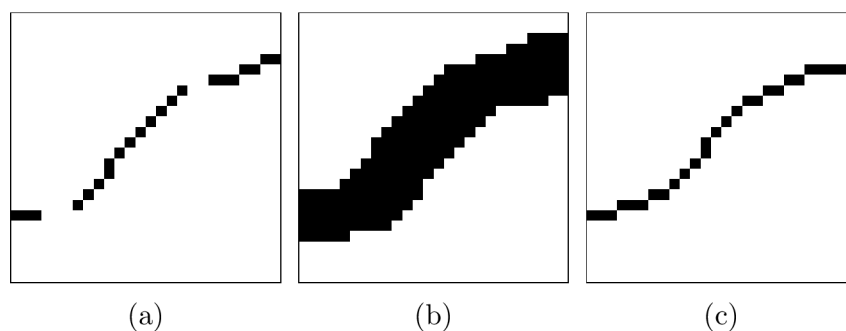
$$o(q) = 1. \quad (4.9)$$

Pri voľbe p sú preferované také body, ktoré ešte žiadnej separovanej krivky nepatria. Ak takéto body neexistujú, podmienka je ignorovaná a nový bod p je volený z celej množiny $S(p)$.



Obr. 4.12: Procedúra **Rozhodni**.

Nakoľko trasovanie je vykonávané na pixeloch obrazu oblúkov a , ktorý neobsahuje body priesečníkov, výsledný obraz o tieto body neobsahuje tiež. Tieto „medzery“ (obrázok 4.13a) sú prepojené opakovaním operácií dilatácie (obrázok 4.13b) a stenčenia (obrázok 4.13c), čím zároveň dochádza k miernemu vyhladeniu výslednej krivky.



Obr. 4.13: (a) detail výstupného obrazu o , (b) po dilatácii a (c) po následnom stenčení

4.2 Estimácia dĺžky

Množina objektových pixelov M_O výstupného obrazu určuje všeobecne viac ako jednu krivku podľa definície 2.1.4. Ak však vyslovíme ľahko pochopiteľný predpoklad, že požadovaná krivka je jednoduchá a má dĺžku $n = |M_O|$, potom je (až na zámenu poradia jej prvkov) určená jednoznačne. Označme túto krivku

$$C = (p_0, \dots, p_n). \quad (4.10)$$

Následkom stenčovania v časti 3.4 táto krivka podhodnocuje skutočnú dĺžku riasového vlákna – vlákno je stenčované rovnomerne po svojom obvode, a jeho pôvodné koncové

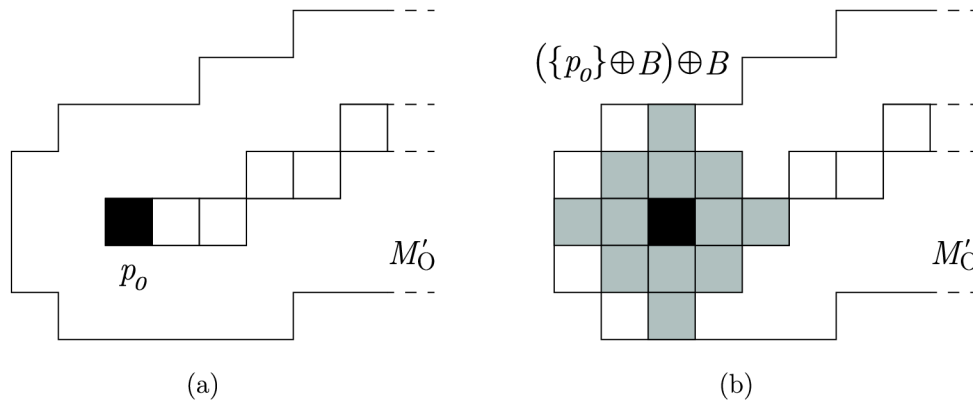
body sú tak posunuté. Krivku C preto na oboch koncoch predĺžime o m bodov, pričom m je najväčšie prirodzené číslo splňujúce

$$\underbrace{\left((\{p_0, p_n\} \oplus B) \dots \oplus B \right)}_{m\text{-krát}} \subseteq M'_O, \quad (4.11)$$

kde M'_O je množina objektových pixelov obrazu získaného morfológickou rekonštrukciou v časti 3.3 a B je štruktúrny element

$$B = \begin{bmatrix} \times & 1 & \times \\ 1 & 1 & 1 \\ \times & 1 & \times \end{bmatrix}. \quad (4.12)$$

Množina koncových bodov $\{p_0, p_n\}$ krivky 4.10, ktorá je následkom vlastností operácie stenčovania podmnožinou množiny objektových pixelov M'_O morfológicky rekonštruovaného obrazu, je teda opakovane dilatovaná do stavu, keď už podmnožinou M'_O nie je. Zaznamenaný je počet opakovaní m , ktorý je pritom vykonaný. Tým je získaná informácia o počte krokov stenčovania, ktoré boli vykonané v rámci predspracovania, *pred dosiahnutím idempotencie* v okolí koncových bodov vlákna.



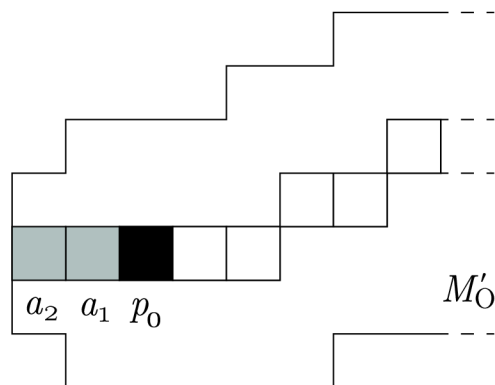
Obr. 4.14: (a) Posunutý koncový bod p_0 krivky C a množina objektových pixelov M'_O obrazu získaného v časti 3.3, (b) 2-krát opakovaná dilatácia množiny $\{p_0\}$.

Tieto body sú potom ku krivke C pridané v smere danom rozdielom súradníc jej posledných dvoch bodov. Upravená krivka je potom;

$$a_m, \dots, a_1, a_0 = p_0, \dots, p_n = b_0, b_1, \dots, b_m, \quad (4.13)$$

kde

$$\begin{aligned} a_0 &= p_0, \\ a_k &= a_{k-1} + (p_0 - p_1), \\ b_0 &= p_n, \\ b_k &= b_{k-1} + (p_n - p_{n-1}), \\ \text{pre } k &= 1, \dots, m. \end{aligned} \quad (4.14)$$



Obr. 4.15: Predĺžená krivka C .

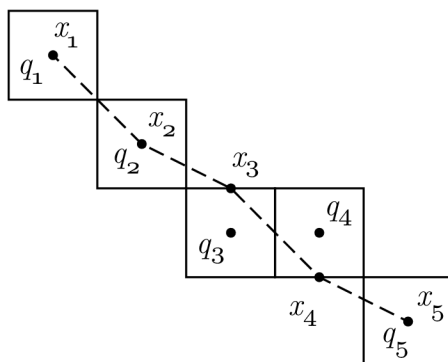
Preznačme body takto predĺzenej digitálnej krivky q_1, \dots, q_N . Mnohé známe metódy estimácie dĺžky kriviek v digitálnom obraze sú založené na ich interpolácii polynómom alebo splajnom – pozorovaním však vidíme, že skúmané vlákna rias pozostávajú z rovných článkov a hladká (analytická) krivka tak nevystihuje ich charakter. Navyše, k určitej miere vyhladenia došlo už pri predchádzajúcich operáciách dilatácie a stenčovania. Na odhad dĺžky je volená metóda upravujúca súradnice bodov q_1, \dots, q_N krivky 4.13 váženým priemerom podľa predpisu

$$x_k = \begin{cases} q_1 & \text{ak } k = 1, \\ q_N & \text{ak } k = N, \\ w(q_{k-1} - q_k) + q_k + w(q_{k+1} - q_k) & \text{inak,} \end{cases} \quad (4.15)$$

kde $w \in [0, 1]$ je váha. Za výsledný odhad ℓ sa potom ukazuje byť vhodné [14] považovať sumu euklidovských vzdialeností

$$\ell = \alpha \sum_{k=1}^n \varrho_2(x_{k-1}, x_k) = \alpha \sum_{k=1}^n \sqrt{(x_k^1 - x_{k-1}^1)^2 + (x_k^2 - x_{k-1}^2)^2}. \quad (4.16)$$

kde $\alpha \in \mathbb{R}$ je konštanta zohľadňujúca mierku pôvodnej snímky.



Obr. 4.16: Body x_k upravujúce súradnice pixelov q_k s váhou $w = 0.5$ podľa vzťahu 4.15.

5 Výsledky

Postupom prezentovaným kapitolách 3 a 4 bola pre každé separované vlákno v snímkach A, B, C a D vypočítaná

- (i) hodnota n vyjadrujúca dĺžku odpovedajúcej jednoduchej digitálnej krivky podľa definície 2.1.4, (t. j. počet pixelov krivky),
- (ii) hodnota l vyjadrujúca odhad dĺžky tejto krivky podľa vzťahu 4.16 pri $\alpha = 1$.

Výsledné hodnoty sú uvedené v tabuľke 5.1. Samotné snímky a vybrané separované vlákna sa nachádzajú v prílohe A.

Tabuľka 5.1: Výsledné hodnoty n a l vlákien rias v snímkach A, B, C a D.

Snímka	Vlákno	n	l
A	I	100	106,03
	II	27	31,21
	III	128	141,72
	IV	1400	1643,55
	V	2578	2852,91
	VI	129	135,21
	VII	177	183,44
	VIII	48	55,1
B	I	837	991,05
	II	100	133,67
	III	45	53,70
	IV	22	21,24
C	I	39	39,18
	II	536	603,75
	III	472	536,99
	IV	2373	2704,90
	V	1417	1646,02
	VI	345	388,16
	VII	315	367,1
	VIII	179	206,27
D	I	349	369,56
	II	691	805,61
	III	1755	1979,46
	IV	605	652,17
	V	112	120,63
	VI	30	37,16

Postupom predspracovania uvedenom v kapitole 3 a originálny algoritmom prezentovaným v kapitole 4 boli zo snímkov A, B, C a D úspešne separované požadované krivky. Vo všetkých z uvedených prípadov pracoval algoritmus spoľahlivo, replikujúc subjektívne vnímanie náväznosti vlákien v pôvodných snímkach. Slabou stránkou algoritmu je jeho

výpočtová náročnosť a požiadavky na operačnú pamäť zariadenia. Aj keď samotná Houghova transformácia náročná nie je, v priebehu algoritmu je vytváraný a ukladaný veľký počet nových obrazov (obraz koncových bodov, obraz priesečníkov, ...). S rastúcimi rozmermi vstupného obrazu potom rýchlo narastá aj časová a výpočtová náročnosť algoritmu. Tomuto problému je však možno predísť – rozmery vstupných obrazov je možné redukovat' prevzorkovaním. Za predpokladu, že týmto prevzorkovaním nedochádza k narušeniu spojitosti objektov na snímke, sa výsledok algoritmu nezmení.

Záver

Práca sa zaoberala problematikou separácie pretínajúcich sa digitálnych kriviek v binárnom obraze. Predstavený bol algoritmus vykonávajúci detekciu a separáciu týchto kriviek. Houghova transformácia, ktorá je nástrojom na detekciu priamok v obraze, bola v algoritme uplatnená novým spôsobom – na aproximáciu „sklonu“ krivky v mieste jej kríženia sa s krivkou inou.

Prezentovaná bola metodika prípravy konkrétnych obrazových dát. Spracovávané boli snímky z mikroskopu zachytávajúce vláknité sinice *Nostoc*, ktoré sa v obraze na viacerých miestach pretínajú. Prostredníctvom lineárnej filtrácie, segmentácie, morfológických operácií a numerických metód boli obrazové dáta pripravené na spracovanie originálnym algoritmom.

V kapitole 1 bol uvedený matematický aparát tvoriaci teoretický základ metód a postupov, ktoré boli v práci použité. Prezentovaná bola primárne teória z oblasti binárnych relácií a metrických priestorov. V kapitole 2 bol definovaný digitálny obraz, jeho vlastnosti a vybrané numerické metódy, ktoré sú v práci použité. V kapitole 3 bol opísaný konkrétny postup, akým boli surové snímky upravené pred tým, než z nich boli separované jednotlivé krivky reprezentujúce vlákna siníc. V kapitole 4 bol predstavený samotný algoritmus, ktorý vykonáva separáciu digitálnych kriviek v binárnom obraze. Výsledky práce boli zhrnuté v kapitole 5.

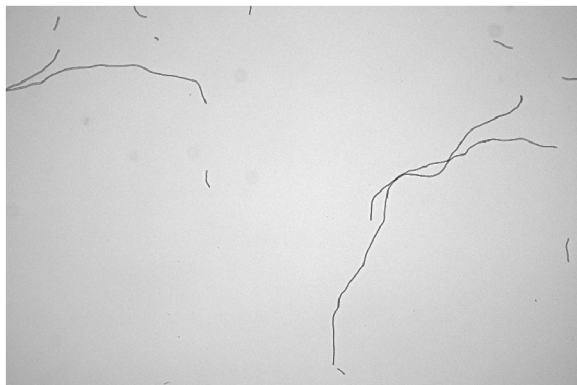
Literatúra

- [1] ŠLAPAL, Josef. *Metody diskrétní matematiky*. Vyd. 2., upr. Brno: Akademické nakladatelství CERM, 2004. ISBN 80-214-2785-x.
- [2] Ústav matematiky FSI VUT. Matematika online. *Množiny, relace mezi množinami* [online]. Brno, 2019 [cit. 2021-03-11]. Dostupné z: <https://mathonline.fme.vutbr.cz/Matematicka-analyzanbspI/sc-1225-sr-1-a-265/default.aspx>
- [3] Ústav matematiky FSI VUT. Matematika online. *Metrické prostory, konvergence v metrickém prostoru* [online]. Brno, 2020 [cit. 2021-03-27]. Dostupné z: <https://mathonline.fme.vutbr.cz/Matematicka-analyzanbspII/sc-1226-sr-1-a-267/default.aspx>
- [4] ŠONKA, Milan, Václav HLAVÁČ a Roger BOYLE. *Image processing, analysis, and machine vision*. 3rd ed. Toronto: Thomson, 2008. ISBN 978-0495082521.
- [5] PRATT, William K. *Digital image processing: PIKS Scientific inside*. 4th ed., Newly updated and rev. ed. Hoboken, N.J.: Wiley-Interscience, c2007. ISBN 978-0-471-76777-0.
- [6] HERMAN, Gabor T. *Geometry of digital spaces*. Boston: Birkhäuser, 1998. ISBN 0-8176-3897-0.
- [7] GONZALEZ, Rafael C. a Richard E. WOODS. *Digital image processing*. 2nd ed. Upper Saddle River: Prentice Hall, c2002. ISBN 02-011-8075-8.
- [8] PIERRE, Soile. *Morphological Image Analysis: Principles and Applications*. 2nd ed. Berlin: Springer, 2002. ISBN 978-3-642-07696-1.
- [9] Picture Thresholding Using an Iterative Selection Method. *IEEE Transactions on Systems, Man, and Cybernetics* [online]. 1978, 8(8), 630-632 [cit. 2021-5-17]. ISSN 0018-9472. Dostupné z: [doi:10.1109/TSMC.1978.4310039](https://doi.org/10.1109/TSMC.1978.4310039)
- [10] VANDEVENNE, Lode. Lode's Computer Graphics Tutorial: Flood Fill. *Lode's Homepage* [online]. [cit. 2021-4-17]. Dostupné z: <https://lodev.org/cgtutor/floodfill.html>
- [11] DUDA, Richard O. a Peter E. HART. Use of the Hough transformation to detect lines and curves in pictures. *Communications of the ACM* [online]. 1972, 15(1), 11-15 [cit. 2021-4-21]. ISSN 0001-0782. Dostupné z: [doi:10.1145/361237.361242](https://doi.org/10.1145/361237.361242)
- [12] HLAVÁČ, Václav. *Matematická morfologie* [online]. Praha: České vysoké učení technické v Praze, 2020 [cit. 2021-4-21]. Dostupné z: <http://people.ciirc.cvut.cz/hlavac/TeachPresCz/11DigZprObr/71-3MatMorpholBinCz.pdf>
- [13] ŠTARHA, Pavel. *Aplikace numerických metod zpracování obrazové informace*. Brno, 2015. Habilitační práce.
- [14] SHAPIRO LINDA, G. a STOCKMAN GEORGE C. *Computer vision*. New Jersey: Prentice-Hall, 2001. ISBN 978-0130307965.

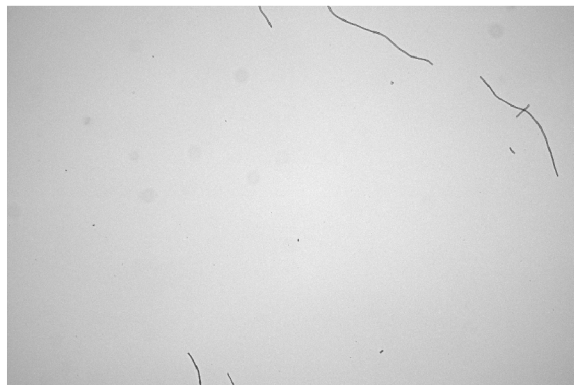
- [15] FICARRA, E., L. BENINI, E. MACII a G. ZUCCHERI. Automated DNA Fragments Recognition and Sizing Through AFM Image Processing. *IEEE Transactions on Information Technology in Biomedicine* [online]. 2005, 9(4), 508-517 [cit. 2021-3-02]. ISSN 1089-7771. Dostupné z: doi:10.1109/TITB.2005.855546

A Snímky

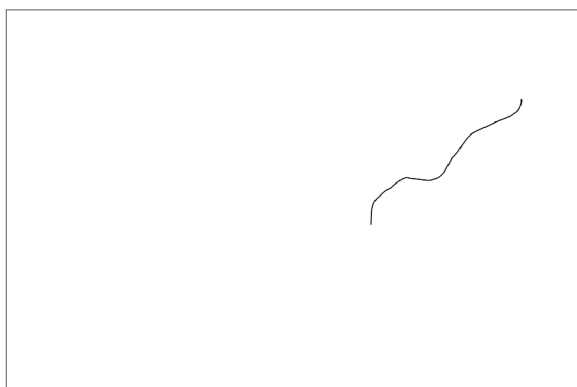
Na nasledujúcich obrázkoch sú vyzobrazené snímky rias, ktoré boli spracovávané, a vybrané vlákna, ktoré boli zo snímok separované.



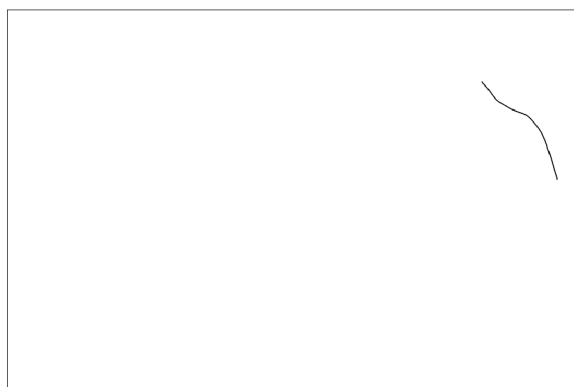
Obr. A.1: Snímka A



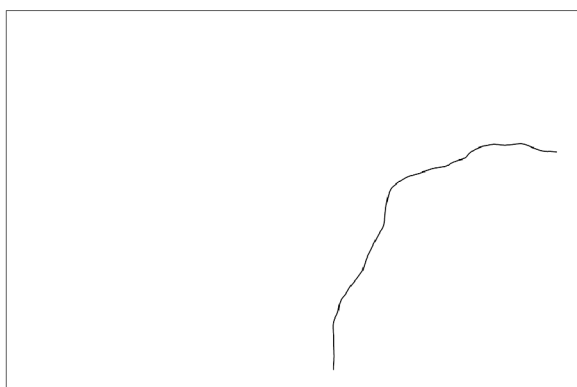
Obr. A.2: Snímka B



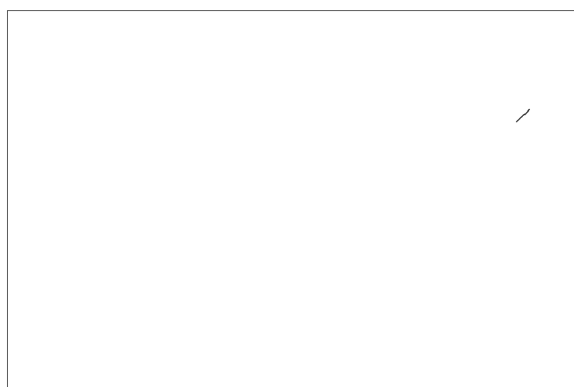
Obr. A.3: Snímka A, vlákno IV



Obr. A.4: Snímka B, vlákno I



Obr. A.5: Snímka A, vlákno IV



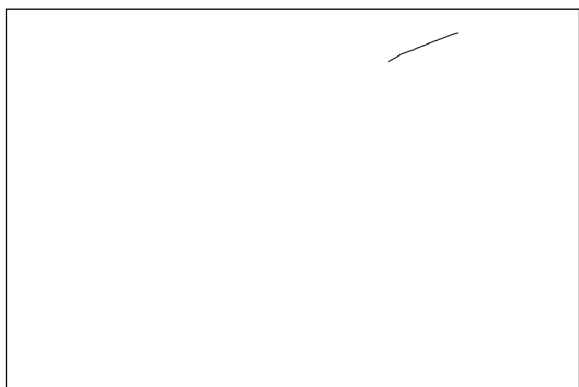
Obr. A.6: Snímka B, vlákno II



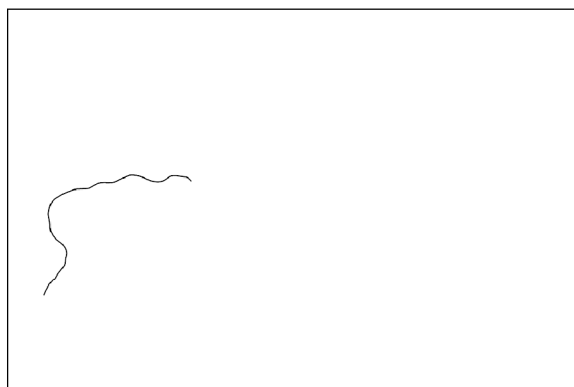
Obr. A.7: Snímka C



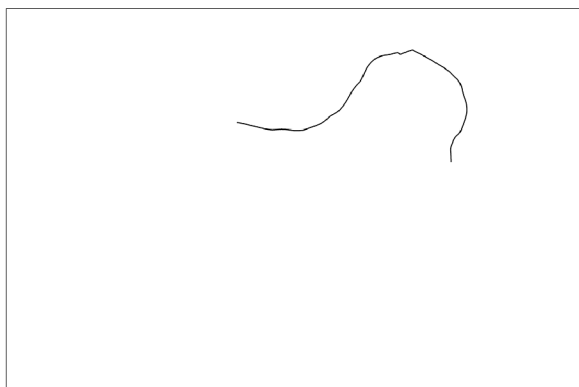
Obr. A.8: Snímka D



Obr. A.9: Snímka C, vlákno II



Obr. A.10: Snímka D, vlákno III



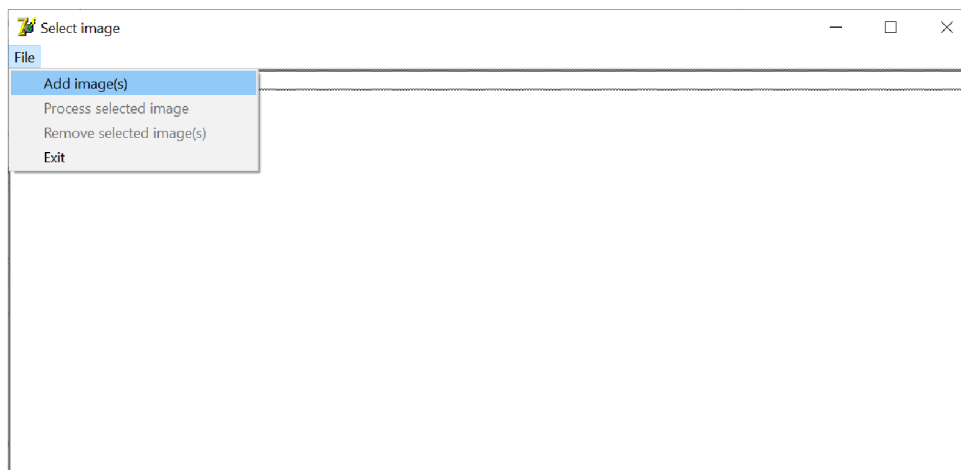
Obr. A.11: Snímka C, vlákno IV



Obr. A.12: Snímka D, vlákno IV

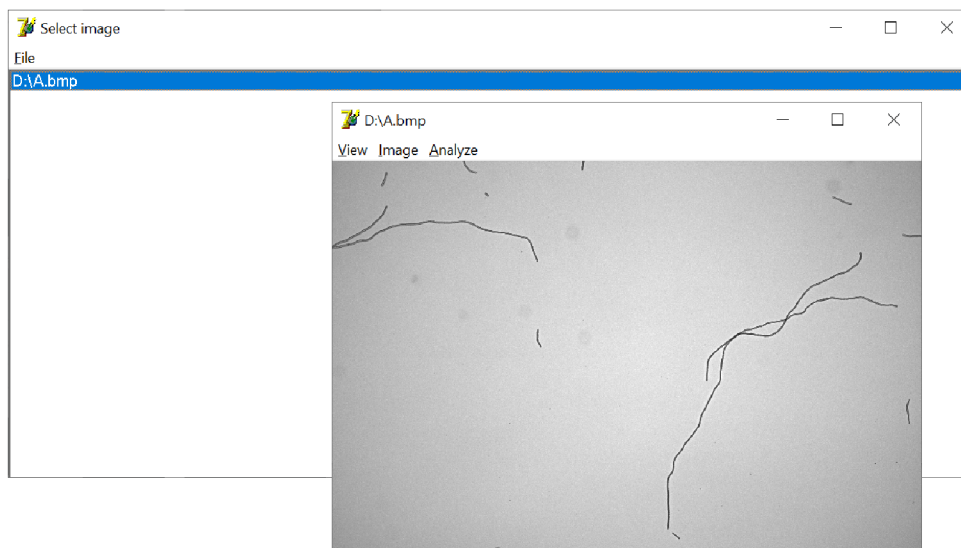
B Opis vytvoreného softvéru

V rámci práce bol vytvorený vlastný softvér vo vývojovom prostredí Delphi 7. Softvér je určený na spracovávanie obrazových dát vo formáte 8-bitových .BMP súborov so šedotónovou paletou, pričom vykonáva všetky operácie a metódy spracovania obrazu uvedené v kapitolách 2, 3 a 4.



Obr. B.1: Hlavné okno softvéru

Obrazové súbory formátu .BMP, ktoré majú byť upravované, je potrebné pridať do zoznamu súborov v hlavnom okno zvolením možnosti **File > Add image(s)**. Pridať je možné viacero súborov zároveň.

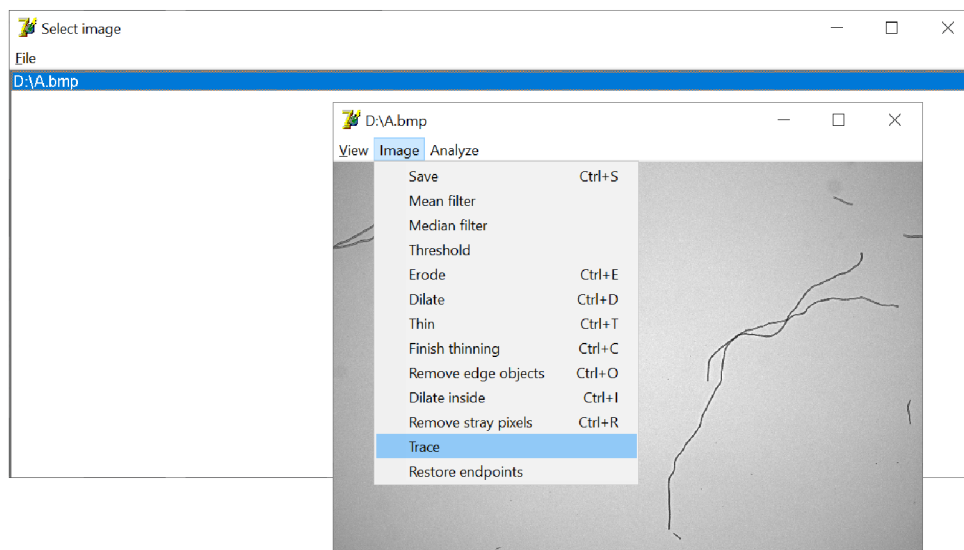


Obr. B.2: Okno s otvoreným súborom

Označením jedného zo súborov v zozname a následným zvolením možnosti **File > Process selected image** sa otvorí nové okno s náhľadom otvoreného .BMP súboru a s ponukou nástrojov na jeho spracovávanie.

Možnosť

- **View > True size** zobrazí spracovávaný obraz v jeho skutočnej veľkosti.
- **View > Fit to window** prispôsobí rozmery spracovávaného obrazu aktuálnym rozmerom okna. Obraz samotný nie je upravovaný, jedná sa len o jeho zobrazenie.
- **View > Fit to window** prispôsobí rozmery okna aktuálnym rozmerom spracovávaného obrazu.
- **Image > Save** ukladá aktuálny stav obrazu do nového, alebo pôvodného .BMP súboru.



Obr. B.3: Ponuka nástrojová
bol

Nástroje vykonávajúce numerické metódy spracovania obrazu sa nachádzajú v záložke **Image**. Možnosť

- **Image > Mean** vyhladzuje obraz lineárnym filtrom kľzavého priemeru s konvolučným jadrom 3×3 .
- **Image > Median** vyhladzuje obraz mediánovým filtrom 3×3 .
- **Image > Threshold** vykonáva prahovanie obrazu.
- **Image > Erode** resp. **Image > Dilate** vykonávajú eróziu, resp. dilatáciu obrazu štruktúrnym elementom 2.6.
- **Image > Thin** vykonáva jednu iteráciu sekvenčného stenčenia postupnosťou štruktúrnych elementov 2.8. Na opakovanie stenčovania je možné využiť klávesovú skratku **CTRL + T**. Stenčovanie je nutné zakončiť možnosťou **Image > Finish thinning**.
- **Image > Remove edge objects** odstraňuje z obrazu objekty dotýkajúce sa jeho okrajov.

- `Image > Dilate inside` vykonáva geodetickú dilatáciu s maskou opísanou v časti 3.3. Pred geodetickou dilatáciou je nutné odstrániť objekty dotýkajúce sa okrajov obrazu.
- `Image > Remove stray pixels` odstraňuje osamotené objektové pixely využitím operácií otvorenia a uzavretia.
- `Image > Trace` vykonáva separáciu kriviek v obraze. Pred použitím možnosti `Trace` je nutné, aby obraz obsahoval iba krivky široké 1 pixel a neobsahoval objekty dotýkajúce sa jeho okrajov. Separácia môže trvať niekoľko desiatok minút a výsledné separované krivky sú ukladané ako samostatné .BMP súbory do priečinku, v ktorom sa nachádza práve spracovávaný .BMP súbor.
- `Image > Restore endpoints` predlžuje separovanú krivku tak, aby jej dĺžka odpovedala dĺžke jej zodpovedajúceho objektu v pôvodnom obraze (vysvetlenie sa nachádza v časti 4.2).

Záložka `Analyze` obsahuje jedinú možnosť `Calculate length`, vyčísľujúcu hodnoty n a l opísané v časti 4.2. Na výpočet týchto hodnôt je nutné, aby otvorený obraz obsahoval iba jednu krivku širokú 1 pixel. Výsledky z kapitoly 5 je možné reprodukovať nasledujúcou postupnosťou operácií:

- (i) vyhladenie nástrojom `Mean filter` alebo `Median filter`
- (ii) prahovanie nástrojom `Threshold`,
- (iii) morfológická rekonštrukcia opakovaným použitím nástroja `Dilate inside`,
- (iv) iteračné stenčenie opakovaným použitím nástroja `Thin`,
- (v) odstránenie patologických štruktúr opakovaným striedaním nástrojov `Dilate` a `Thin`,
- (vi) odstránenie okrajových objektov nástrojom `Remove edge objects`,
- (vii) separácia kriviek nástrojom `Trace`,
- (viii) spojenie a vyhladenie separovaných kriviek opakovaným striedaním nástrojov `Dilate` a `Thin`,
- (ix) predĺženie kriviek nástrojom `Restore endpoints` a konečný výpočet nástrojom `Calculate length`.

K práci sú okrem samotného softvéru priložené snímky A, B, C a D vo formáte .BMP. Rozmery priložených snímok sú zredukované prevzorkovaním metódou bilineárnej interpolácie.