

SLOVENSKÁ TECHNICKÁ UNIVERZITA V BRATISLAVE
STAVEBNÁ FAKULTA

Reg. No.: SvF-5343-87813

Segmentácia mračien bodov
z terestrického laserového skenera s využitím vyvíjajúcich sa kriviek

Diplomová práca

Študijný program:	Aplikovaná matematika
Študijný odbor:	Matematika
Školiace pracovisko:	Katedra matematiky a deskriptívnej geometrie
Konzultant:	Ing. Mgr. Lukáš Tomek, PhD.

Bratislava 2021

Bc. Branislav Beran



ZADANIE DIPLOMOVEJ PRÁCE

Študent: **Bc. Branislav Beran**
ID študenta: 87813
Študijný program: matematicko-počítačové modelovanie
Študijný odbor: matematika
Vedúci práce: Ing. Mgr. Lukáš Tomek, PhD.

Názov práce: **Segmentácia mračien bodov z terestrického laserového skenera s využitím vyvíjajúcich sa kriviek**

Jazyk, v ktorom sa práca vypracuje: slovenský jazyk

Špecifikácia zadania:

Skenovanie pomocou terestrického laserového skenera je metóda priestorového merania scenérií. Výstupom je mračno bodov, v ktorom poznáme priestorové súradnice každého bodu a obvykle aj jeho farbu a intenzitu odrazeného laserového lúča. V prípade skenovania stavebných objektov množstvo bodov leží v rovinách (steny, stropy, podlahy), ktoré je možné odhadnúť pomocou regresných metód a vysegmentovať tak body, ktoré s istou presnosťou ležia v rovine. Takto získané segmentácie rovín často okrem želaných bodov (napr. ležiacich na stene) obsahujú aj ďalšie body ležiace v tej istej rovine (napr. dvere, či elektrické zásuvky a vypínače), ktoré je potrebné zo segmentácie odstrániť.

A tu prichádzajú na scénu vyvíjajúce sa krivky. Z počiatočnej malej kružnice ležiacej v segmentovanej rovine bude krivka expandovať, pričom rýchlosť vývoja môže byť riadená napríklad hustotou mračna, farbou bodov, intenzitou odrazených lúčov či kolmou vzdialenosťou od regresnej roviny. Vývoj krivky sa zastaví, keď krivka ohraničí želané body. Hlavným cieľom práce je vyvinúť a v softvéri MATLAB implementovať algoritmus na takúto segmentáciu.

Riešenie zadania práce od: 08. 09. 2020

Dátum odovzdania práce: 13. 05. 2021

Bc. Branislav Beran
študent

Ing. Marek Macák, PhD.
vedúci pracoviska

prof. RNDr. Karol Mikula, DrSc.
garant študijného programu

Abstract

The work deals with the curve segmentation of planar point clouds, scanned by a terrestrial laser scanner, which usually represent the walls of buildings. The work describes a mathematical model of evolving curves, which are used to segment objects lying in planar point clouds. The evolution of curves is controlled by properties of the scanned walls, such as color, intensity or curvature measure of the wall. We discretize this model using the finite volume method and the semi-implicit scheme. In addition to the mathematical model for curve evolution, we also describe other methods and algorithms needed for the whole curve evolution process, such as the method for inserting initial curves, aggregating point cloud properties using fuzzy logic operators and the method of topological changes to merge and split curves. We implement algorithms in the MATLAB programming environment and demonstrate their functionality on real data.

Keywords: evolving curves, segmentation, point cloud

Abstrakt

Práca sa zaoberá krivkovou segmentáciou rovinných mračien bodov naskenovaných terestrickým laserovým skenerom, ktoré obvykle predstavujú steny budov. V práci je popísaný matematický model vyvíjajúcich sa kriviek, ktoré slúžia na segmentáciu objektov ležiacich v rovinných mračnách. Vývoj kriviek je riadený vlastnosťami naskenovaných stien, ako napríklad farba, intenzita alebo miera krivosti steny. Tento model diskretizujeme použitím metódy konečných objemov a semi-implicitnej schémy. Okrem matematického modelu na vývoj krivky v práci popisujeme aj ďalšie metódy a algoritmy potrebné pre celý proces vývoja kriviek, ako napríklad metódu na vkladanie počiatočných kriviek, spájanie vlastností mračna bodov pomocou fuzzy logických operátorov a metódu topologických zmien na spájanie a rozdeľovanie kriviek. Algoritmy implementujeme v programovacom prostredí MATLAB a ich funkčnosť demonštrujeme na reálnych dátach.

Kľúčové slová: vývoj kriviek, segmentácia, mračno bodov

Predhovor

Táto práca je súčasťou projektu APVV-18-0247 s názvom Automatizácia kontroly elektronickej dokumentácie stavieb s využitím inovatívnych technológií zberu údajov a virtuálnych modelov. Cieľom tohoto projektu je vypracovať metodiku kontroly skutočného vyhotovenia stavieb v prostredí BIM (Building Information Modeling) a rozšíriť funkcionality BIM o nástroje na automatizovanú kontrolu kvality realizovaného diela, na báze automatizovanej tvorby 3D modelov z údajov získaných terestrickým laserovým skenerom.

BIM je parametrický objektovo-orientovaný dátový 3D model stavby, ktorý obsahuje informácie o konštrukciách, prvkoch a ich parametroch.

Proces celého projektu je naskenovať budovy terestrickým laserovým skenerom, následne dáta vo forme mračna bodov spracovať v dvojkrokovej segmentácii pomocou algoritmov vyvinutých v programovacom prostredí MATLAB. Následne sa segmenty porovnávajú s objektami z BIM modelu.

V prvom kroku segmentácie sa hľadajú regresné roviny z naskenovaných dát. Takéto rovinné segmenty bodov predstavujú najmä steny, stropy a podlahy skenovanej scenérie.

Po prvom kroku segmentácie sa často stáva, že vysegmentované rovinné mračná obsahujú ďalšie objekty, ako napríklad zásuvky, vypínače, dvere a podobne. Primárnou úlohou je získať body ležiace priamo na stene, preto sa v druhom kroku segmentácie hľadajú body, ktoré prislúchajú iným objektom, aby sme ich vedeli oddeliť od bodov steny.

Táto diplomová práca sa zaoberá druhým krokom segmentácie a to segmentácia objektov z rovinných mračien bodov pomocou vyvíjajúcich sa kriviek.

V kapitole 1 vysvetľujeme, ako funguje terestrický laserový skener, ako vyzerá výstup zo skenera a opisujeme hlavnú myšlienku segmentácie naskenovaných dát.

V kapitole 2 popisujeme význam jednotlivých členov matematického modelu na vývoj kriviek. Jedným z členov je tangenciálna rýchlosť, ktorú počítame využitím tangenciálnej redistribúcie a popisujeme ju v sekcii 2.2. Druhým členom je normálová rýchlosť popísaná v sekcii 2.3, ktorá zahŕňa funkciu homogenity, hranové detektory a krivostnú regularizáciu.

Kapitola 3 je zameraná na diskretizáciu matematického modelu z kapitoly 2 pomocou metódy konečných objemov a použitím semi-implicitnej metódy na časovú diskretizáciu. V kapitole ďalej popisujeme spôsob reprezentácie mračna bodov, ktorý využívame vo výpočtoch, spájanie vlastností mračna na vývoj krivky, metódu na spájanie a rozdeľovanie kriviek počas vývoja a spôsob vkladania počiatočných kriviek.

V kapitole 4 popisujeme aplikáciu algoritmov a metód na niekoľkých numerických experimentoch.

Na záver sa chcem poďakovať Ing. Mgr. Lukášovi Tomekovi, PhD. za cenné pripomienky a množstvo času, ktorý mi venoval počas písania práce. Vďaka patrí aj mojím rodičom a sestre za nesmiernu podporu počas celého štúdia.

Bratislava 13. 5. 2021

Branislav Beran

Obsah

Predhovor	5
1 Úvod	9
1.1 Terestrický laserový skener	9
1.2 Vstupné dáta	9
1.3 Segmentácia regresnými metódami na získanie odhadov rovinných podmračien	10
1.4 Segmentácia rovinných podmračien pomocou vyvíjajúcich sa kriviek	11
2 Matematický model	13
2.1 Zostavenie matematického modelu	13
2.2 Tangenciálna rýchlosť	14
2.3 Normálová rýchlosť	15
2.3.1 Krivostná regularizácia	15
2.3.2 Normálová rýchlosť riadená vonkajším vplyvom	16
2.3.3 Funkcia homogenity	17
2.3.4 Hranový detektor	18
2.4 Úprava modelu pomocou Frenetových-Serretových rovníc	18
3 Diskretizácia	20
3.1 Diskretizácia rovnice pre evolúciu krivky	20
3.1.1 Metóda konečných objemov	20
3.1.2 Časová diskretizácia - Semi-Implicitná schéma	21
3.2 Sieť rovinného mračna bodov	25
3.2.1 Odchýlka normál	26
3.2.2 Výpočet hranového detektora na pixelovej reprezentácii vlastností rovinného mračna	26
3.3 Diskretizácia tangenciálnej rýchlosti	27
3.4 Diskretizácia normálovej rýchlosti	29
3.4.1 Diskretizácia krivostnej regularizácie	29
3.4.2 Diskretizácia normálovej rýchlosti riadenej vonkajším vplyvom	29
3.5 Spájanie hranových detektorov a funkcií homogenity	31
3.5.1 Fuzzy logika	31
3.5.2 Spájanie hranových detektorov	32
3.5.3 Spájanie funkcií homogenity	34
3.6 Topologické zmeny	34
3.6.1 Pridávanie a odoberanie bodov	34
3.6.2 Rozdeľovanie krivky	35
3.6.3 Spájanie kriviek	36
3.7 Vkládanie počiatočných kriviek	38
3.8 Zastavovacie kritérium	40
3.9 Postup celého algoritmu	40

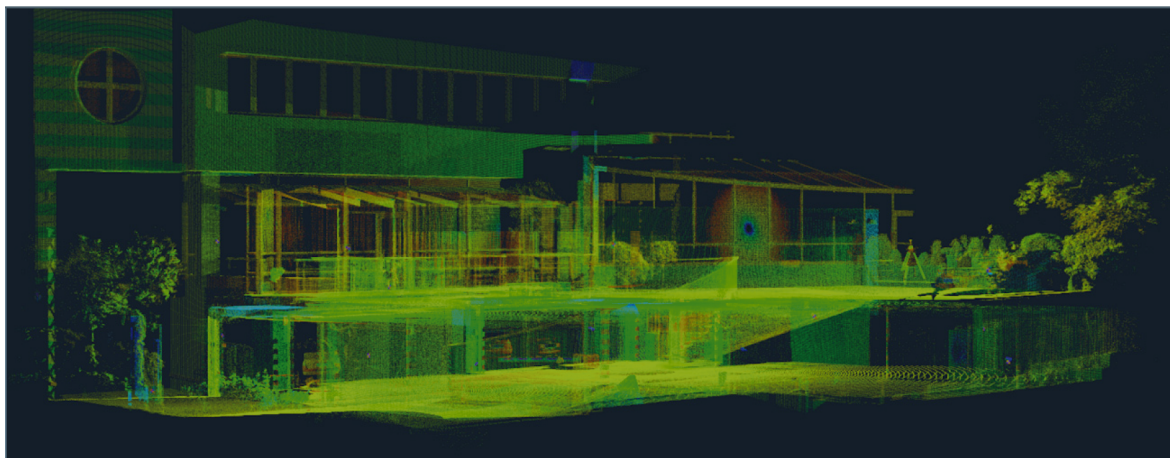
4	Numerické experimenty	42
4.1	Stena s dverami	42
4.2	Stena s plagátmi a obkladom	48
4.3	Predná stena haly	54
4.4	Poschodová budova	58
5	Záver	63

Kapitola 1

Úvod

1.1 Terestrický laserový skener

V súčasnosti je laserové skenovanie jednou z najefektívnejších metód na zber priestorových údajov a tvorbu 3D modelov. Terestrický laserový skener je zariadenie, ktoré určuje súradnice bodov ležiacich na povrchu meraných objektov. Výsledkom skenu sú dáta vo forme mračna bodov, pričom namerané body sú rozmiestnené v rasti, definovanom pravidelnými uhlovými rozostupmi v horizontálnom aj vertikálnom smere. Toto zariadenie dokáže naskenovať milióny bodov počas niekoľkých sekúnd, a tým umožňuje tvorbu komplexnejších modelov s detailnejšími údajmi o naskenovaných objektoch. V našej práci využívame skeny budov získané práve touto technológiou.



Obr. 1.1: Vizualizácia naskenovaného mračna bodov. Tento obrázok sme použili z knihy [3].

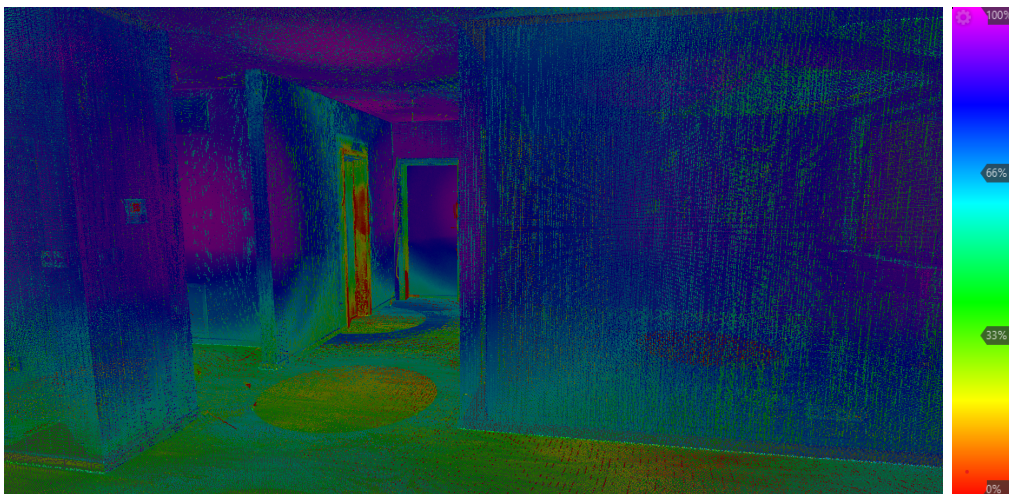
1.2 Vstupné dáta

Ako je vyššie spomínané, dáta z laserového skenera podrobne skenujú scenériu a výstupom je mračno bodov, pričom pre každý bod máme informáciu o polohe, farbe bodu a o intenzite odrazeného laserového lúča. Pomocou polohy bodov sa hľadajú rovinné segmenty, ktoré budú tvoriť vstupné dáta do ďalšieho kroku, a to segmentácie objektov z rovinných mračen bodov. Následne na rovinných mračnách bodov využijeme údaje o vlastnostiach v jednotlivých bodoch na segmentáciu objektov rôznych farieb, alebo aj materiálov, ktoré sa môžu líšiť v intenzite odrazených lúčov. Pomocou polohy bodov vieme získať niekoľko ďalších informácií, ktoré nám môžu pomôcť v segmentácii objektov na rovinných mračnách bodov. Takouto informáciou môže byť lokálna odchýlka normál bodov, ktorá súvisí s krivosťou skenovaného

povrchu a môže byť užitočná v segmentácii objektov s menej výrazným rozdielom farby a intenzity od okolia. Pre lepšiu predstavu o týchto dátach si ukážeme pohľad na 3D model bytu naskenovaného laserovým skenerom.



Obr. 1.2: Vykreslenie mračna bodov zafarbených podľa farby.



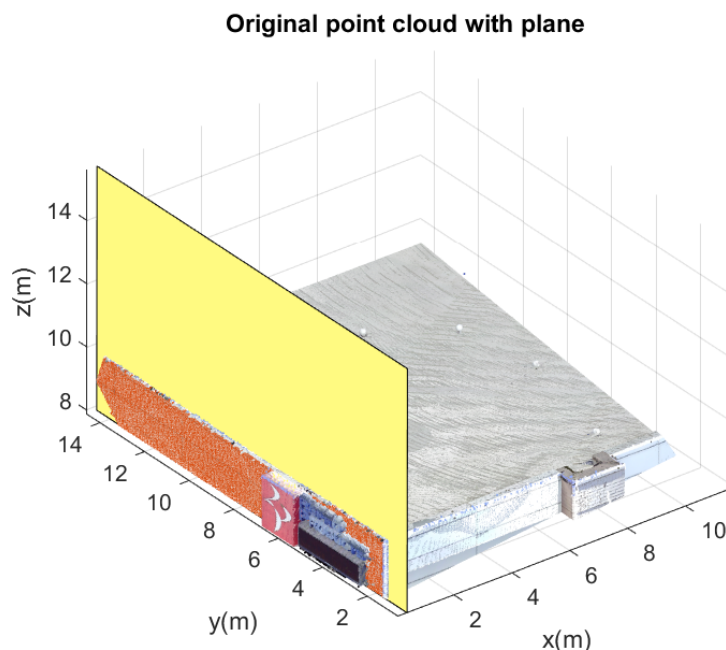
Obr. 1.3: Vykreslenie mračna bodov zafarbených podľa intenzity odrazených lúčov.

1.3 Segmentácia regresnými metódami na získanie odhadov rovinných podmračien

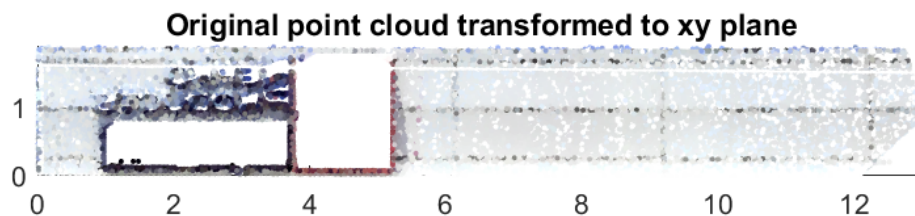
Prvým krokom segmentácie je segmentácia rovinných podmračien, takže takých podmnožín mračna, ktorých body ležia približne v jednej rovine. Táto časť segmentácie nie je náplňou tejto práce, ale výsledné segmenty tvoria vstupné dáta pre analýzu, ktorou sa v našej práci zaoberáme, a to segmentáciu objektov z rovinných mračien bodov, preto si len stručne priblížime, aká metóda sa tu využíva, a ako vyzerá výsledok tejto segmentácie.

Na získavanie rovinných segmentov sa používa algoritmus M-estimated Sample Consensus (MSAC), ktorý je modifikáciou algoritmu Random Sample Consensus (RANSAC). Algoritmus RANSAC náhodne vyberá 3 body mračna, následne sa týmito tromi bodmi preloží rovina a zisťuje sa počet bodov, ktorých kolmá vzdialenosť od roviny je menšia, ako určená hodnota. Ak je počet týchto bodov dostatočne veľký vzhľadom k celkovému počtu bodov mračna, tak

tieto body budú tvoriť jeden zo segmentov. Body ležiace v rovine sú následne transformované do lokálnych 2D súradníc, ako to môžeme vidieť v obrázku 1.5



Obr. 1.4: Znáznornenie segmentácie roviny z mračna bodov.

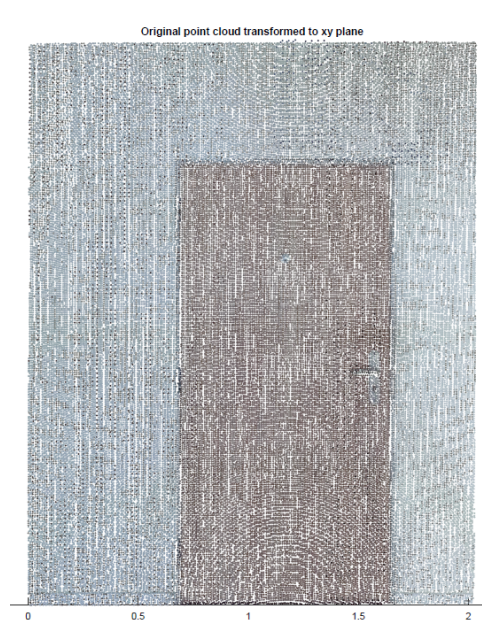


Obr. 1.5: Vysegmentované mračno transformované do lokálnych 2D súradníc.

Získavanie rovinných segmentov sa opakuje, kým nezostane len malé množstvo bodov vzhľadom k celkovému počtu bodov mračna.

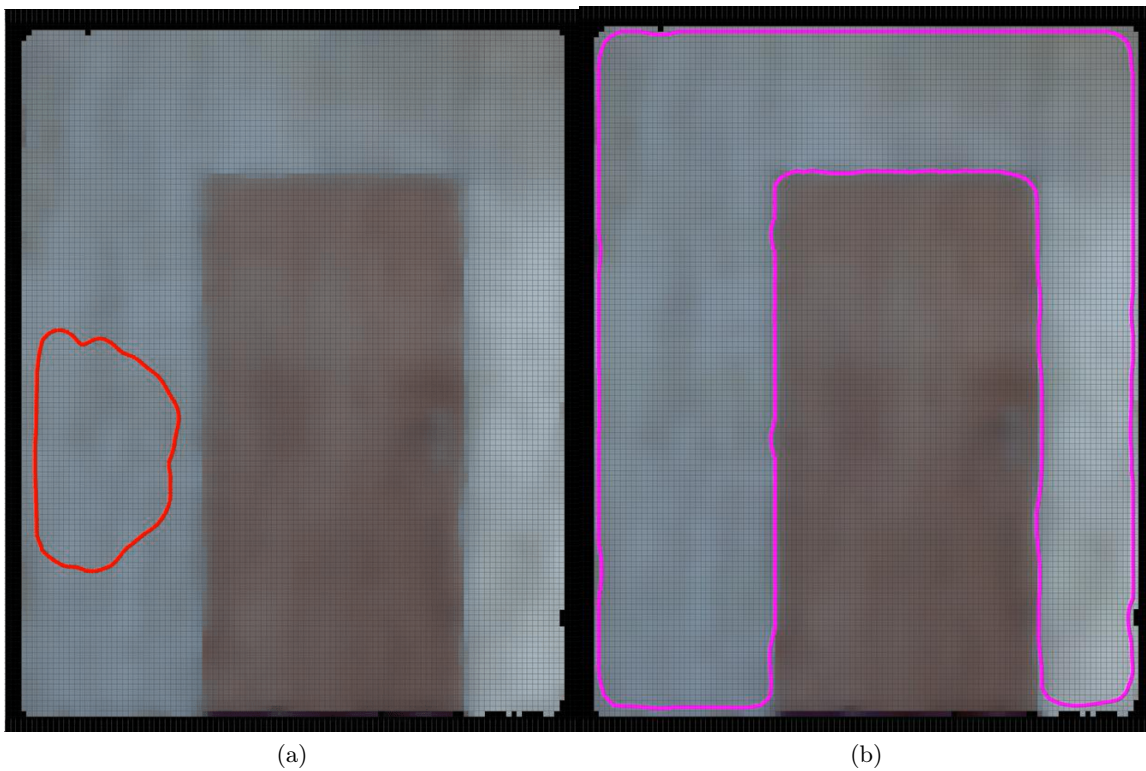
1.4 Segmentácia rovinných podmračien pomocou vyvíjajúcich sa kriviek

Vysegmentované roviny, ako napríklad steny, podlahy a stropy, budeme ďalej analyzovať. Naším cieľom bude nájsť tie body mračna, ktoré prislúchajú k stene, a to tak, že odstránime rôzne objekty, ako napríklad dvere, vypínače, elektrické zásuvky a podobne. Na segmentáciu budeme využívať vyvíjajúce sa krivky, ktorých pohyb bude riadený hlavne vlastnosťami mračna bodov, ako napríklad farba, intenzita alebo odchýlka normál. Dôležitou súčasťou bude aj aplikácia hranových detektorov, ktoré nám pomôžu odhaliť hrany objektov ležiacich v rovinách a tým ovplyvnia aj vývoj krivky. Na obrázku 1.6 môžeme vidieť mračno bodov, so stenou, v ktorej ležia dvere.



Obr. 1.6: Mračno bodov vyfarbené farebnými kanálmi R, G a B.

Do tohoto mračna budeme vkladať krivku, ktorá sa bude vyvíjať a počas vývoja obklopí dvere ležiace v stene. Pre lepšiu predstavu môžeme vidieť na obrázku 1.7 krivku vo vývoji (vľavo) a vo finálnom štádiu (vpravo) po tom čo obklopí dvere a zastaví.



Obr. 1.7: Evolúcia krivky.

Kapitola 2

Matematický model

2.1 Zostavenie matematického modelu

Majme uzavretú pozitívne orientovanú krivku $\gamma_0 : I \rightarrow \mathbb{R}^2$. Krivku γ_0 budeme vyvíjať v čase, čo znamená, že pre každé $u \in I$ bude existovať trajektória pohybu v časovom intervale $T = [t_0, t_N]$. Našou neznámou je funkcia

$$\gamma : I \times T \rightarrow \mathbb{R}^2, \quad (2.1)$$

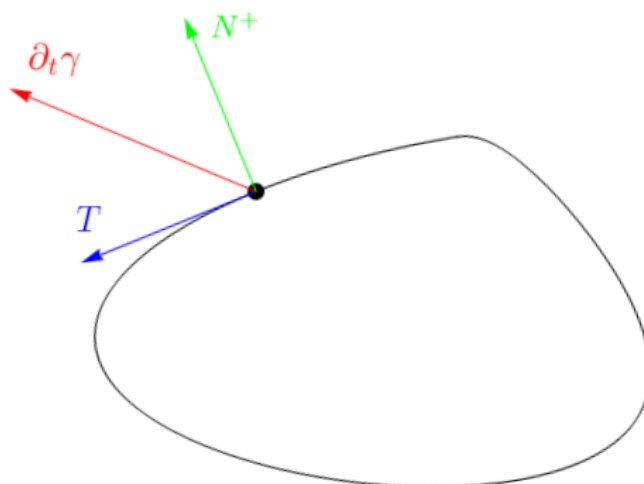
pričom platí, že $\gamma(I, t_0) = \gamma_0$, a teda krivka γ_0 bude našou počiatočnou podmienkou. Keďže chceme riadiť vývoj krivky, tak nás bude zaujímať, ako sa bude krivka meniť v čase. To vieme matematicky vyjadriť ako časovú deriváciu krivky γ , alebo fyzikálne ako rýchlosť vývoja krivky v , čo vieme zapísať nasledujúcim vzťahom

$$\partial_t \gamma(u, t) = v(u, t). \quad (2.2)$$

Na pravej strane rovnice vystupuje funkcia rýchlosti vývoja, ktorú vieme rozdeliť na tangenciálnu zložku $\alpha(u, t)T(u, t)$, pričom T je dotykový vektor a normálovú zložku $\beta(u, t)N^+(u, t)$, pričom $N^+(u, t)$ je kladne orientovaný normálový vektor

$$\partial_t \gamma(u, t) = \alpha(u, t)T(u, t) + \beta(u, t)N^+(u, t), \quad (2.3)$$

kde $\alpha(u, t)$ je funkcia tangenciálnej rýchlosti a $\beta(u, t)$ funkcia normálovej rýchlosti.



Obr. 2.1: Znázornenie vektorov na krivke γ .

2.2 Tangenciálna rýchlosť

V tejto sekcii používame poznatky o tangenciálnej redistribúcii z článku [4]. Keďže tangenciálna rýchlosť nemení tvar krivky, ale iba posúva body pozdĺž krivky, tak pre nás nie je až taká zásadná, ako rýchlosť v smere normály. Na druhú stranu, môže byť veľmi užitočná pri numerickom riešení vývoja kriviek. Počas vývoja krivky sa môže stať, že veľká časť intervalu I by bola zobrazená na veľmi malý úsek krivky, čo by v analytickom riešení nemusel byť veľký problém, ale v prípade numerického riešenia by to mohlo vytvárať veľké nepresnosti a prispieť k nestabilite numerickej schémy. Preto budeme na výpočet funkcie pre tangenciálnu rýchlosť $\alpha(u, t)$ používať tangenciálnu redistribúciu. Majme $u_1, u_2 \in I$ a $t_1, t_2 \in T$, potom účelom tangenciálnej redistribúcie je, aby pomer vzdialenosti bodov $\gamma(u_1, t_1)$ a $\gamma(u_2, t_1)$ po krivke voči dĺžke celej krivky bol rovnaký ako vzdialenosť bodov $\gamma(u_1, t_2)$ a $\gamma(u_2, t_2)$ po krivke voči dĺžke celej krivky. Takáto tangenciálna redistribúcia zachováva relatívne dĺžky na krivke γ , čo vieme matematicky zapísať nasledovne

$$\frac{\int_{u_1}^{u_2} \|\partial_u \gamma^t\| du}{L(\gamma^t)} = C, \quad C \in \mathbb{R}, \quad (2.4)$$

preto musí platiť diferenciálna rovnica

$$\partial_t \left(\frac{\|\partial_u \gamma^t\|}{L(\gamma^t)} \right) = 0. \quad (2.5)$$

V našom prípade budeme používať asymptoticky rovnomerné rozdelenie bodov, ktoré sa asymptoticky približuje k rovnomernej vzdialenosti medzi susediacimi bodmi krivky, namiesto relatívnych dĺžok na krivke. Tieto dve metódy sa odlišujú v pravej strane rovnice (2.5), pričom pre asymptoticky rovnomerné rozdelenie bodov vyzerá matematický zápis nasledovne

$$\partial_t \left(\frac{\|\partial_u \gamma^t\|}{L(\gamma^t)} \right) = \left(1 - \frac{\|\partial_u \gamma\|}{L(\gamma^t)} \right) \omega. \quad (2.6)$$

Rozderivovaním ľavej strany dostávame rovnicu

$$\frac{\partial_t \|\partial_u \gamma^t\| L(\gamma^t) - \partial_t L(\gamma^t) \|\partial_u \gamma^t\|}{L(\gamma^t)^2} = \left(1 - \frac{\|\partial_u \gamma\|}{L(\gamma^t)} \right) \omega. \quad (2.7)$$

Z vlastností vyvíjajúcich sa kriviek vieme, že platia vzťahy

$$\partial_t \|\partial_u \gamma\| = -\|\partial_u \gamma\| \kappa^\pm \beta + \partial_u \alpha, \quad (2.8)$$

$$\partial_t L(\gamma^t) = - \int_{\text{Im} \gamma} \kappa^\pm \beta ds. \quad (2.9)$$

Aplikovaním týchto vzťahov dostávame rovnicu v tvare

$$\frac{(-\|\partial_u \gamma^t\| \kappa^\pm \beta + \partial_u \alpha) L(\gamma^t) - \|\partial_u \gamma^t\| (- \int_{\text{Im} \gamma} \kappa^\pm \beta ds)}{L(\gamma^t)^2} = \left(1 - \frac{\|\partial_u \gamma\|}{L(\gamma^t)} \right) \omega, \quad (2.10)$$

a túto rovnicu si prenasobíme $\frac{L(\gamma)}{\|\partial_u \gamma\|}$ a následne vyjadríme člen $\frac{\partial_u \alpha}{\|\partial_u \gamma\|}$

$$\frac{\partial_u \alpha}{\|\partial_u \gamma\|} = \kappa^\pm \beta - \frac{1}{L(\gamma^t)} \int_{\text{Im} \gamma} \kappa^\pm \beta ds + \left(1 - \frac{L(\gamma^t)}{\|\partial_u \gamma\|} \right) \omega. \quad (2.11)$$

Ak si krivku γ reparametrizujeme použitím funkcie dĺžky

$$s(u) = \int_0^u \|\gamma(\hat{u}, t)\| d\hat{u}, \quad (2.12)$$

získame prirodzenú reparametrizáciu, ktorej myšlienkou je „kráčať po krivke“ konštantnou rýchlosťou $\|\partial_u \gamma(u, t)\| = 1$. Vzťah medzi funkciou tangenciálnej rýchlosti parametrizovanou prirodzenou parametrizáciou a inou parametrizáciou je nasledovný

$$\bar{\alpha}(s(u)) = \alpha(u). \quad (2.13)$$

Potom deriváciou tohoto vzťahu dostávame rovnicu

$$\partial_u \alpha = \frac{\partial \bar{\alpha}}{\partial s} \cdot \frac{\partial s}{\partial u} = \partial_s \bar{\alpha} \|\partial_u \gamma^t\|. \quad (2.14)$$

Použitím tohoto vzťahu dostávame vzťah pre tangenciálnu redistribúciu krivky γ

$$\partial_s \bar{\alpha} = \kappa^\pm \beta - \frac{1}{L(\gamma^t)} \int_{\text{Im} \gamma} \kappa^\pm \beta ds + \left(\frac{L(\gamma^t)}{\|\partial_u \gamma\|} - 1 \right) \omega. \quad (2.15)$$

Aby táto diferenciálna rovnica mala jednoznačné riešenie, tak musíme zadať nejakú konkrétnu hodnotu α_0 v jednom z bodov krivky.

2.3 Normálová rýchlosť

Na výstavbu modelu normálovej rýchlosti sme sa inšpirovali článkom [5]. Rýchlosť v smere pozitívne orientovanej normály bude určovať funkcia $\beta(u, t)$, ktorú si v našom modeli rozdelíme na súčet dvoch členov

$$\beta(u, t) = \delta(u, t) \kappa^\pm + \beta_E(u, t), \quad (2.16)$$

kde prvý člen je krivostná regularizácia a hovoríme o nej v sekcii 2.3.1 a druhý člen je normálová rýchlosť riadená vonkajším vplyvom, ktorú popisujeme v sekcii 2.3.2.

2.3.1 Krivostná regularizácia

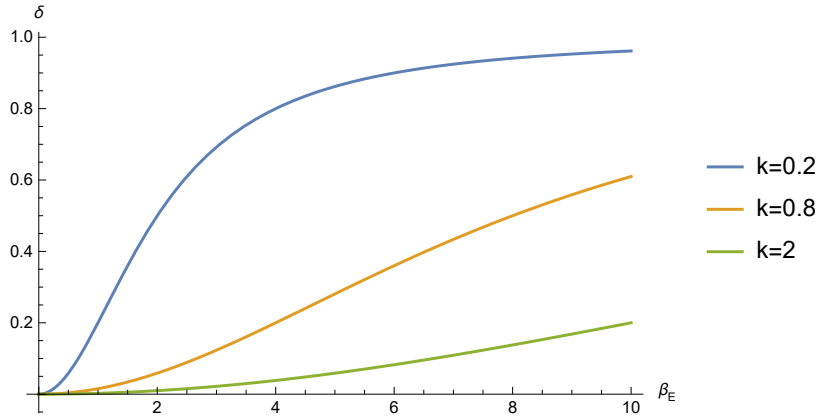
Prvý člen z rovnice (2.16): $\delta(u, t) \kappa^\pm$ sa nazýva aj krivostná regularizácia. Účel tohoto člena je zhladzovať krivku hlavne v oblastiach, kde je vysoká krivosť. Použitím tohoto člena zabraňujeme krivke nadobudnúť príliš ostré tvary, ktoré by mohli byť problematické pri numerickom riešení diferenciálnej rovnice nášho modelu pre evolúciu krivky. Funkciu $\delta(u, t)$ počítame nasledovne

$$\delta(u, t) = (1 - \lambda(t)) \delta_{\text{global}} \cdot \delta_{\text{local}}(u, t) \cdot g_1(\gamma(u, t)), \quad (2.17)$$

pričom δ_{global} je hodnota, ktorá je fixná počas celého vývoja krivky a platí, že čím je vyššia, tým väčšie je zhladzovanie krivostnou regularizáciou. Funkcie $\lambda(t)$ a $g_1(\gamma(u, t))$ súvisia s hranovými detektormi, o ktorých detailne hovoríme v sekcii 2.3.2 a 3.4, ale ich úlohou v krivostnej regularizácii je znižovať zhladzovanie, keď sa krivka začne približovať k hrane nejakého objektu ležiaceho v regresnej rovine, aby sa krivka vedela priblížiť aj do rohov objektov. Funkciu $\delta_{\text{local}}(u, t)$ počítame takto

$$\delta_{\text{local}}(u, t) = 1 - \frac{1}{1 + \left(\frac{|\beta_E(u, t)|}{k \cdot \max(\beta_E^t)} \right)^2}, \quad (2.18)$$

pričom $\max(\beta_E^t)$ je maximálna hodnota funkcie β_E v čase t . Je to lokálna váha krivostnej regularizácie a pomocou nej riadime mieru zhladzovania krivky a teda, čím je väčšia normálová rýchlosť závislá od vonkajších vplyvov β_E^t , tak tým väčšia je aj miera zhladzovania v danom bode krivky. Na obrázku 2.2, ako ovplyvňuje hodnotu funkcie δ_{local} parameter k .

Obr. 2.2: Funkcia δ_{local} pre rôzne hodnoty parametra k .

2.3.2 Normálová rýchlosť riadená vonkajším vplyvom

Druhý člen z rovnice (2.16): $\beta_E(u, t)$ bude ovplyvňovať pohyb krivky v smere pozitívne orientovaného normálového vektora v závislosti od mračna bodov naskenovaného terestrickým laserovým skenerom a budeme ho počítat nasledovným vzťahom

$$\beta_E(u, t) = (1 - \lambda(t))g_2(\gamma(u, t)) + \lambda(t)((-\nabla g_1(\gamma(u, t))) \cdot N^+). \quad (2.19)$$

Rovnica (2.19) pozostáva z dvoch členov, ktoré riadia vývoj krivky. Úlohou prvého člena $g_2(\gamma(u, t))$ je expandovať krivku, až kým sa nepriblíži k hrane. Túto funkciu budeme počítat nasledovne

$$g_2(\gamma(u, t)) = H(\gamma(u, t))g_1(\gamma(u, t)). \quad (2.20)$$

Použitím tejto rovnice vieme rovnicu (2.19) zapísať takto

$$\beta_E(u, t) = (1 - \lambda(t))H(\gamma(u, t))g_1(\gamma(u, t)) + \lambda(t)((-\nabla g_1(\gamma(u, t))) \cdot N^+). \quad (2.21)$$

Funkcia $H(\gamma(u, t))$ sa nazýva funkcia homogenity a jej účelom je expandovať krivku v miestach, ktoré majú podobné hodnoty vlastností mračna ako na oblasti prejdenej krivkou od začiatku vývoja. Naopak, ak je miesto, v ktorom leží bod krivky odlišný od oblasti, ktorú krivka prešla, tak tento člen spomalí expanziu v tomto bode krivky. Ako túto funkciu počítame, si vysvetlíme v sekcii 2.3.3. Funkcia $g_1(\gamma(u, t))$ je hranový detektor, ktorý vystupuje v oboch častiach rovnice (2.19), a tiež ho podrobnejšie popíšeme v sekcii 2.3.4. V prvom člene rovnice táto funkcia pomáha k spomaleniu vývoja krivky na nájdených hranách a v ich okolí.

Úlohou druhého člena rýchlosti z rovnice (2.19): $(-\nabla g_1(\gamma(u, t))) \cdot N^+$, je pritiahnúť krivku bližšie k nájdenej hrane. Tento člen začne pôsobiť hlavne vtedy, keď sa rýchlosť vývoja krivky spomalí. Parameter $\lambda(t)$ slúži na prepínanie medzi týmito dvoma členmi a jeho hodnota je z intervalu $[0, 1]$, pričom platí, že čím je väčšia hodnota tohoto parametra, tým väčší vplyv má druhý člen rovnice, ktorý priťahuje krivku k hranám a čím je menšia tým viac pôsobí expanzný člen.

Funkcia $\lambda(t)$ slúži na prepínanie medzi expanzným členom a členom, ktorý priťahuje body ku hranám. Táto funkcia má skoro počas celého vývoja krivky hodnotu rovnú 0.5, čo znamená, že oba členy majú rovnakú váhu na celkovú hodnotu $\beta_E(u, t)$, avšak keď je krivka po celej dĺžke na hranách, tak sa jej rýchlosť spomalí a vtedy nastavíme funkciu $\lambda(t)$ na hodnotu 1, čo znamená, že expanzný člen prestane pôsobiť a pôsobiť bude iba druhý člen rovnice, ktorý priťahuje body k hranám.

2.3.3 Funkcia homogenity

Funkcia homogenity má za úlohu expandovať krivku a počíta sa použitím vlastností mračna bodov. Pre každú vlastnosť počítame túto funkciu separátne a nakoniec ich spájame do jednej výslednej funkcie homogenity podľa postupu popísaného v sekcii 3.5.3. V tejto podkapitole si ukážeme, ako počítame funkciu homogenity pre jednu z vlastností.

Majme funkciu $I(x)$, ktorú si môžeme predstaviť, ako interpoláciu hodnôt jednej z vlastností na bodoch mračna, napríklad intenzita odrazených laserových lúčov. Ďalej si zdefinujme množinu $\Omega(t)$, ktorá predstavuje oblasť, po ktorej prešla krivka od počiatku vývoja zjednotenú s vnútorom počiatkovej krivky, potom majme funkciu $d(\gamma(u, t))$, ktorá bude vyzeráť nasledovne

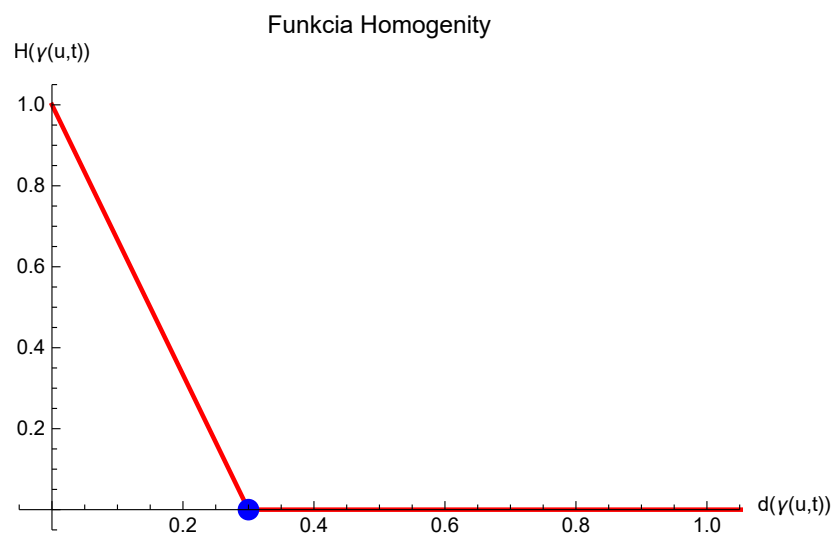
$$d(\gamma(u, t)) = \left| I(\gamma(u, t)) - \frac{1}{|\Omega(t)|} \int_{\Omega(t)} I d\Omega \right|. \quad (2.22)$$

Funkcia $I(\gamma(u, t))$ predstavuje hodnotu vlastnosti v bode $\gamma(u, t)$. Nás zaujíma rozdiel vlastnosti v bode $\gamma(u, t)$ od priemernej hodnoty tejto vlastnosti na oblasti $\Omega(t)$, čomu zodpovedá druhá časť rovnice $\frac{1}{|\Omega(t)|} \int_{\Omega(t)} I d\Omega$. Čím je tento rozdiel väčší, tým je hodnota vlastnosti v bode krivky odlišnejšia od oblasti, ktorú krivka prešla od počiatku a podľa toho budeme nastavovať rýchlosť vývoja krivky, preto si našu funkciu homogenity navrhujeme nasledovne

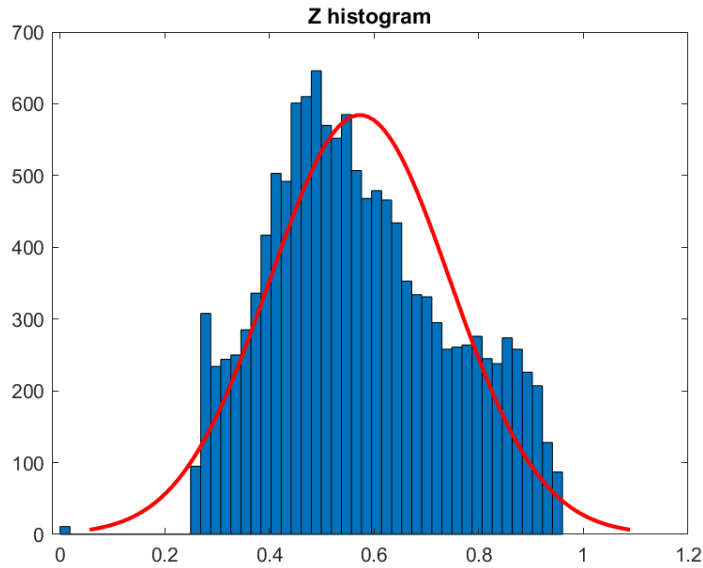
$$H(\gamma(u, t)) = \begin{cases} 1 - \frac{d(\gamma(u, t))}{\mu}, & d(\gamma(u, t)) \leq \mu \\ 0, & \text{inak} \end{cases}. \quad (2.23)$$

Vizualizácia tejto funkcie je na obrázku 2.3. Prahovú hodnotu μ počítame z histogramu vlastnosti mračna bodov, ktorý je pre intenzitu vykreslený na obrázku 2.4. Najprv si vypočítame histogram obrázku a následne distribučnú funkciu normálneho rozdelenia a jej štandardnú odchýlku σ_I . Hodnotu parametra μ počítame ako násobok štandardnej odchýlky

$$\mu = 1.5 \cdot \sigma_I. \quad (2.24)$$



Obr. 2.3: Funkcia homogenity.



Obr. 2.4: Histogram.

2.3.4 Hranový detektor

Funkciu $g_1(\gamma(u, t))$, ktorá vystupuje v oboch častiach rovnice (2.19), a aj v rovnici pre krivostnú regularizáciu (2.17), budeme počítat z rovníc pre hranový detektor. Tento člen odhaľuje hrany objektov v regresnej rovine, a tým pomáha pri segmentácii. Túto funkciu budeme počítat nasledovne

$$g_1(x) = (G_{\sigma_1} * g(s, k))(x), \quad (2.25)$$

a teda ako konvolúciu funkcie $g(s, k)$ s Gaussovským konvolučným jadrom a zaujímať nás budú hlavne hodnoty v miestach, kde leží krivka. Táto funkcia bude vracat hodnoty blízke jednej v miestach, kde sa nenachádzajú hrany a hodnoty blízke nule v miestach kde sa hrany nachádzajú. Funkcia $g(s, k)$ z tejto rovnice má nasledujúci tvar

$$g(s, k) = \frac{1}{1 + ks^2}. \quad (2.26)$$

Parameter k ovplyvňuje to, aké budú hrany výrazné a platí, že čím je hodnota tohoto parametra vyššia, tým sú hrany výraznejšie. Premenná s sa počíta pomocou noriem gradientov z predhľadenej funkcie I

$$s = \|\nabla(G_{\sigma_0} * I)\|. \quad (2.27)$$

2.4 Úprava modelu pomocou Frenetových-Serretových rovníc

S použitím tvaru (2.16) pre normálovú rýchlosť v rovnici (2.3) vieme náš model zapísať nasledovne

$$\partial_t \gamma = \alpha T + \beta_E N^+ + \delta \kappa^\pm N^+. \quad (2.28)$$

Zatiaľ sme nekonkretizovali parametrizáciu krivky, a keďže budeme rovnicu transformovať Frenetovými-Serretovými rovnicami, ktorých podmienkou je prirodzená parametrizácia, tak budeme uvažovať krivku γ prirodzene parametrizovanú parametrom s . Frenetove-Serretove

rovnice popisujú, ako sa mení vektor T a N^+ pozdĺž krivky.

$$\begin{aligned}\partial_s T &= \kappa^\pm N^+, \\ \partial_s N^+ &= -\kappa^\pm T.\end{aligned}\tag{2.29}$$

Na rovnicu (2.28) vieme aplikovať 1. uvedenú Frenetovu-Serretovu rovnicu zo vzťahov (2.29) a dostávame rovnicu v nasledujúcom tvare

$$\partial_t \gamma = \alpha T + \beta_E N^+ + \delta \partial_s T.\tag{2.30}$$

Okrem Frenetových-Serretových rovníc, môžeme použiť aj vzťah pre dotykový vektor T krivky γ

$$T = \partial_s \gamma.\tag{2.31}$$

Pozitívne orientovaný normálový vektor N^+ získame otočením dotykového vektora T o 90° proti smeru hodinových ručičiek, a preto vieme predchádzajúcu rovnicu využiť na vyjadrenie vzťahu medzi krivkou γ a N^+

$$N^+ = T^\perp = (\partial_s \gamma)^\perp.\tag{2.32}$$

Po aplikovaní vzťahov (2.31) a (2.32), bude náš matematický model v tvare

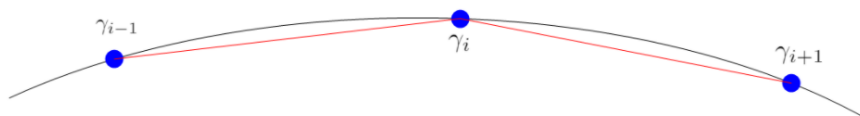
$$\partial_t \gamma - \alpha \partial_s \gamma = \beta_E (\partial_s \gamma)^\perp + \delta \partial_s^2 \gamma.\tag{2.33}$$

Kapitola 3

Diskretizácia

3.1 Diskretizácia rovnice pre evolúciu krivky

Náš matematický model pre vývoj krivky popisuje rovnica (2.33). Najprv si našu krivku diskretizujeme na m bodov, pričom i -ty bod budeme označovať γ_i . Polohu bodu γ_i , získame



Obr. 3.1: Diskretizácia krivky.

získame nasledovne

$$\gamma_i = \gamma((i-1)\Delta u, t), \quad i = 1, \dots, m, \quad (3.1)$$

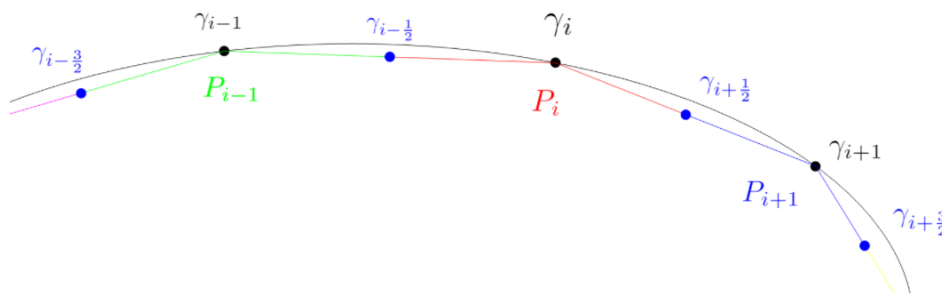
pričom Δu je krok na intervale $I = [0, 1]$, ktorý si rozdelíme na m rovnako dlhých častí, tým pádom bude dĺžka

$$\Delta u = \frac{1}{m}. \quad (3.2)$$

Keďže krivka je uzavretá, tak dodefinujeme $\gamma_0 = \gamma_m$ a $\gamma_{m+1} = \gamma_1$. Na diskretizáciu rovnice (2.33) budeme používať metódu konečných objemov.

3.1.1 Metóda konečných objemov

Podľa metódy konečných objemov budeme uvažovať okolie diskretizačných bodov krivky, ako konečné objemy, pričom pre i -ty bod γ_i majme konečný objem $P_i = [\gamma_{i-\frac{1}{2}}, \gamma_{i+\frac{1}{2}}]$, pozri obrázok 3.2.



Obr. 3.2: Diskretizácia krivky.

Ďalším krokom metódy konečných objemov je integrácia rovnice (2.33) na oblasti konečného objemu P_i , po ktorej dostávame nasledujúcu rovnicu

$$\int_{\gamma_{i-\frac{1}{2}}}^{\gamma_{i+\frac{1}{2}}} \partial_t \gamma \, ds - \int_{\gamma_{i-\frac{1}{2}}}^{\gamma_{i+\frac{1}{2}}} \alpha \partial_s \gamma \alpha \, ds = \int_{\gamma_{i-\frac{1}{2}}}^{\gamma_{i+\frac{1}{2}}} \delta \partial_s^2 \gamma \, ds + \int_{\gamma_{i-\frac{1}{2}}}^{\gamma_{i+\frac{1}{2}}} \beta_E (\partial_s \gamma)^\perp \, ds \quad (3.3)$$

Funkcie, ktoré budeme po diskretizácii považovať za konštantné na danom konečnom objeme, vyjme pred integrály a rovnica nášho modelu na konečnom objeme P_i bude vyzeráť takto

$$\partial_t \gamma_i \int_{\gamma_{i-\frac{1}{2}}}^{\gamma_{i+\frac{1}{2}}} ds - \alpha_i \int_{\gamma_{i-\frac{1}{2}}}^{\gamma_{i+\frac{1}{2}}} \partial_s \gamma \, ds = \delta_i \int_{\gamma_{i-\frac{1}{2}}}^{\gamma_{i+\frac{1}{2}}} \partial_s^2 \gamma \, ds + \beta_{E,i} \int_{\gamma_{i-\frac{1}{2}}}^{\gamma_{i+\frac{1}{2}}} (\partial_s \gamma)^\perp \, ds. \quad (3.4)$$

Na výpočet hraničných bodov oblasti konečného objemu používame aproximácie

$$\gamma_{i-\frac{1}{2}} \approx \frac{\gamma_{i-1} + \gamma_i}{2}, \quad (3.5)$$

$$\gamma_{i+\frac{1}{2}} \approx \frac{\gamma_i + \gamma_{i+1}}{2} \quad (3.6)$$

a veľkosť konečného objemu budeme aproximovať vzťahom

$$|P_i| \approx \frac{h_{i-1} + h_i}{2}, \quad (3.7)$$

kde h_i je vzdialenosť medzi bodmi γ_i a γ_{i+1} , takže

$$h_i = \|\gamma_i - \gamma_{i+1}\|. \quad (3.8)$$

Po aplikácii Newton-Leibnitzovho vzťahu dostávame

$$\partial_t \gamma_i [1]_{\gamma_{i-\frac{1}{2}}^{\gamma_{i+\frac{1}{2}}}} - \alpha_i [\gamma]_{\gamma_{i-\frac{1}{2}}^{\gamma_{i+\frac{1}{2}}}} = \delta_i [\partial_s \gamma]_{\gamma_{i-\frac{1}{2}}^{\gamma_{i+\frac{1}{2}}}} + \beta_{E,i} \left([\gamma]_{\gamma_{i-\frac{1}{2}}^{\gamma_{i+\frac{1}{2}}}} \right)^\perp. \quad (3.9)$$

$$\partial_t \gamma_i \frac{h_{i-1} + h_i}{2} - \alpha_i \left(\gamma_{i-\frac{1}{2}} - \gamma_{i+\frac{1}{2}} \right) = \delta_i [\partial_s \gamma]_{\gamma_{i-\frac{1}{2}}^{\gamma_{i+\frac{1}{2}}}} + \beta_{E,i} \left(\gamma_{i-\frac{1}{2}} - \gamma_{i+\frac{1}{2}} \right)^\perp \quad (3.10)$$

. Po použití centrálnych konečných diferencií v tvare

$$\partial_s \gamma_{i-\frac{1}{2}} = \frac{\gamma_{i+1} - \gamma_i}{h_i}, \quad (3.11)$$

$$\partial_s \gamma_{i-\frac{1}{2}} = \frac{\gamma_i - \gamma_{i-1}}{h_{i-1}}, \quad (3.12)$$

dostávame priestorovo diskretizovanú rovnicu nášho matematického modelu

$$\partial_t \gamma_i \frac{h_{i-1} + h_i}{2} - \alpha_i \left(\frac{\gamma_{i+1} - \gamma_{i-1}}{2} \right) = \delta_i \left(\frac{\gamma_{i+1} - \gamma_i}{h_i} - \frac{\gamma_i - \gamma_{i-1}}{h_{i-1}} \right) + \beta_{E,i} \left(\frac{\gamma_{i+1} - \gamma_{i-1}}{2} \right)^\perp. \quad (3.13)$$

3.1.2 Časová diskretizácia - Semi-Implicitná schéma

Na diskretizáciu časovej derivácie použijeme doprednú konečnú diferenciu

$$\partial_t \gamma_i \approx \frac{\gamma_i^{n+1} - \gamma_i^n}{\Delta t}. \quad (3.14)$$

V tejto diskretizácii značí jej horný index časový krok, kde n je starý časový krok a $n+1$ je nový časový krok, pričom n začína od hodnoty 1 a ide až po maximálny počet časových iterácií, alebo kým vývoj nezastaví zastavovacie kritérium, o ktorom si povieme neskôr. Použitím (3.14)

na (3.13) dostávame

$$\begin{aligned} & \frac{\gamma_i^{n+1} - \gamma_i^n}{\Delta t} \frac{h_{i-1}^n + h_i^n}{2} - \alpha_i^n \left(\frac{\gamma_{i+1}^{n+1} - \gamma_{i-1}^{n+1}}{2} \right) \\ &= \delta_i^n \left(\frac{\gamma_{i+1}^{n+1} - \gamma_i^{n+1}}{h_i^n} - \frac{\gamma_i^{n+1} - \gamma_{i-1}^{n+1}}{h_{i-1}^n} \right) + \beta_{E,i}^n \left(\frac{\gamma_{i+1}^n - \gamma_{i-1}^n}{2} \right)^\perp. \end{aligned} \quad (3.15)$$

Využitím princípov semi-implicitnej schémy vystupujú v rovnici členy z rôznych časových krokov. Všimnime si, že krivku γ uvažujeme v $n + 1$ časovom kroku v 2. člene ľavej strany rovnice, v ktorom vystupuje tangenciálna rýchlosť a podobne aj v prvom člene pravej strany, kde vystupuje krivostná regularizácia, avšak v druhom člene pravej strany, v ktorej vystupuje normálová rýchlosť riadená vonkajším vplyvom vystupuje krivka γ v časovom kroku n . Rovnicu preusporiadame a dostávame

$$\begin{aligned} & - \left(\frac{\delta_i^n}{h_{i-1}^n} - \frac{\alpha_i^n}{2} \right) \gamma_{i-1}^{n+1} + \left(\frac{h_{i-1}^n - h_i^n}{2\Delta t} + \delta_i^n \left(\frac{1}{h_i^n} + \frac{1}{h_{i-1}^n} \right) \right) \gamma_i^{n+1} - \left(\frac{\delta_i^n}{h_i^n} + \frac{\alpha_i^n}{2} \right) \gamma_{i+1}^{n+1} \\ &= \frac{h_{i-1}^n + h_i^n}{2\Delta t} \gamma_i^n + \beta_{E,i}^n \left(\frac{\gamma_{i+1}^n - \gamma_{i-1}^n}{2} \right)^\perp. \end{aligned} \quad (3.16)$$

Takto preusporiadaná rovnica obsahuje na ľavej strane iba γ z časového kroku $n + 1$ a na pravej z časového kroku n . Pred tým, ako si zapíšeme túto sústavu rovníc v maticovom tvare, si pre zjednodušenie zavedieme nasledujúce označenia

$$\begin{aligned} A_i^n &= \frac{\delta_i^n}{h_{i-1}^n} - \frac{\alpha_i^n}{2}, \\ B_i^n &= \frac{h_{i-1}^n - h_i^n}{2\Delta t} + \delta_i^n \left(\frac{1}{h_i^n} + \frac{1}{h_{i-1}^n} \right), \\ C_i^n &= \frac{\delta_i^n}{h_i^n} + \frac{\alpha_i^n}{2}, \\ D_i^n &= \frac{h_{i-1}^n + h_i^n}{2\Delta t} \gamma_i^n + \beta_{E,i}^n \left(\frac{\gamma_{i+1}^n - \gamma_{i-1}^n}{2} \right)^\perp. \end{aligned} \quad (3.17)$$

Potom rovnicu (3.16) vieme zapísať ako

$$-A_i^n \gamma_{i-1}^{n+1} + B_i^n \gamma_i^{n+1} - C_i^n \gamma_{i+1}^{n+1} = D_i^n \quad (3.18)$$

a náš systém rovníc bude v maticovom zápise vyzerat nasledovne

$$\begin{bmatrix} B_1^n & -C_1^n & 0 & \dots & 0 & -A_1^n \\ -A_2^n & B_2^n & -C_2^n & 0 & \dots & 0 \\ 0 & -A_3^n & B_3^n & -C_3^n & 0 & \\ \vdots & & & \ddots & & \\ -C_m^n & \dots & & -A_m^n & B_m^n & \end{bmatrix} \cdot \begin{bmatrix} \gamma_1^{n+1} \\ \vdots \\ \gamma_m^{n+1} \end{bmatrix} = \begin{bmatrix} D_1^n \\ \vdots \\ D_m^n \end{bmatrix}. \quad (3.19)$$

Takéto systémy sa nazývajú cyklicky trojdiagonálne a najefektívnejšia metóda na riešenie takýchto systémov je Tomasov algoritmus a jeho modifikácie. Tomasov algoritmus, ale nefunguje stabilne, ak matica systému nie je diagonálne dominantná, čo v našom prípade nie je bezpodmienečne. Aby bola táto matica diagonálne dominantná, musí platiť

$$|B_i^n| \geq |-A_i^n| + |-C_i^n|, \quad i = 1, \dots, m. \quad (3.20)$$

Keďže všetky členy B_i^n sú vždy kladné, môžeme pre tento člen absolútnu hodnotu odstrániť. Po spätnom dosadení z (3.17), dostávame vzťah

$$\frac{h_{i-1}^n + h_i^n}{2\Delta t} + \delta_i^n \left(\frac{1}{h_i^n} + \frac{1}{h_{i-1}^n} \right) \geq \left| \frac{\delta_i^n}{h_{i-1}^n} - \frac{\alpha_i^n}{2} \right| + \left| \frac{\delta_i^n}{h_i^n} + \frac{\alpha_i^n}{2} \right|. \quad (3.21)$$

Táto podmienka nemusí byť vždy splnená, zachrániť to môžeme správnou voľbou časového kroku Δt , pričom po úpravách vieme vyjadriť podmienku, ktorú časový krok musí spĺňať pre každú rovnicu z nášho systému

$$\Delta t \leq \frac{h_{i-1}^n + h_i^n}{2 \left(\left| \left(\frac{\delta_i^n}{h_{i-1}^n} - \frac{\alpha_i^n}{2} \right) \right| + \left| \left(\frac{\delta_i^n}{h_i^n} - \frac{\alpha_i^n}{2} \right) \right| - \delta_i^n \left(\frac{1}{h_i^n} + \frac{1}{h_{i-1}^n} \right) \right)}. \quad (3.22)$$

Aby sme zaistili, že túto podmienku spĺňa každá rovnica systému (3.19), tak by sme museli vypočítať pravú stranu tejto nerovnosti pre každú rovnicu systému a následne podľa minima z týchto hodnôt určiť časový krok Δt . To by bolo ale časovo náročné, a preto našu semi-implicitnú schému upravíme použitím IIOE metódy.

IIOE metóda

Inflow implicit outflow explicit v skratke IIOE, je metóda, ktorá nám pomôže vyriešiť problém s nestabilitou schémy v prípade voľby veľkého časového kroku Δt . Skôr, ako si ukážeme úpravu schémy IIOE metódou, pozrime sa na to, kde vzniká problém, kvôli ktorému nieje schéma bezpodmienečne stabilná. Problematická je tangenciálna rýchlosť α_i^n . V prípade, že $\alpha_i^n = 0$, tak by nerovnica (3.21) mala tvar

$$\frac{h_{i-1}^n + h_i^n}{2\Delta t} + \delta_i^n \left(\frac{1}{h_i^n} + \frac{1}{h_{i-1}^n} \right) \geq \left| \frac{\delta_i^n}{h_{i-1}^n} \right| + \left| \frac{\delta_i^n}{h_i^n} \right|. \quad (3.23)$$

Keďže zlomky v absolútnych hodnotách na pravej strane nerovnice dosahujú iba kladné hodnoty, tak absolútne hodnoty môžeme odstrániť, a všimnime si, že druhý člen ľavej strany nerovnice je potom rovný pravej strane nerovnice, preto ju vieme upraviť nasledovne

$$\frac{h_{i-1}^n + h_i^n}{2\Delta t} \geq 0. \quad (3.24)$$

Keďže všetky členy ľavej strany sú väčšie ako 0, tak je táto podmienka splnená pre ľubovoľne veľký kladný časový krok Δt a tým pádom je schéma bezpodmienečne stabilná. Bezpodmienečnú stabilitu by sme chceli mať aj v prípade, že $\alpha_i^n \neq 0$, preto sa vráťme k rovnici (3.13), kde urobíme úpravu člena, v ktorom vystupuje α_i^n

$$-\alpha_i \left(\frac{\gamma_{i+1} - \gamma_{i-1}}{2} \right) = \frac{1}{2}(-\alpha_i)(-\gamma_{i-1}) + \frac{1}{2}\alpha_i(-\gamma_{i+1}) + \frac{1}{2}(-\alpha_i)\gamma_i + \frac{1}{2}\alpha_i\gamma_i. \quad (3.25)$$

Časť rovnice písaná modrou farbou je len pripočítanie nuly. Pravú stranu upravíme do nasledovného tvaru

$$\frac{1}{2}(-\alpha_i)(-\gamma_{i-1}) + \frac{1}{2}\alpha_i(-\gamma_{i+1}) + \frac{1}{2}(-\alpha_i)\gamma_i + \frac{1}{2}\alpha_i\gamma_i = \frac{1}{2}(-\alpha_i)(\gamma_i - \gamma_{i-1}) + \frac{1}{2}\alpha_i(\gamma_i - \gamma_{i+1}). \quad (3.26)$$

Ďalej si zdefinujeme pomocné označenia,

$$\begin{aligned} b_{i-\frac{1}{2}}^{in} &= \max(-\alpha_i, 0) \geq 0, \\ b_{i-\frac{1}{2}}^{out} &= \min(-\alpha_i, 0) \leq 0, \\ b_{i+\frac{1}{2}}^{in} &= \max(\alpha_i, 0) \geq 0, \\ b_{i+\frac{1}{2}}^{out} &= \min(\alpha_i, 0) \leq 0. \end{aligned} \quad (3.27)$$

V tomto prípade je dôležité, ktoré členy sú nezáporné, a ktoré nekladné. Pomocou týchto definícií môžeme upraviť rovnicu (3.26) nasledovne

$$\frac{1}{2}(-\alpha_i)(\gamma_i - \gamma_{i-1}) + \frac{1}{2}\alpha_i(\gamma_i - \gamma_{i+1}) = \frac{1}{2}(b_{i-\frac{1}{2}}^{in} + b_{i-\frac{1}{2}}^{out})(\gamma_i - \gamma_{i-1}) + \frac{1}{2}(b_{i+\frac{1}{2}}^{in} + b_{i+\frac{1}{2}}^{out})(\gamma_i - \gamma_{i+1}). \quad (3.28)$$

Po takejto úprave rovnice člena α_i dostávame rovnicu (3.13) v takomto tvare

$$\begin{aligned} \partial_t \gamma_i \frac{h_{i-1} - h_i}{2} + \frac{1}{2}(b_{i-\frac{1}{2}}^{in} + b_{i-\frac{1}{2}}^{out})(\gamma_i - \gamma_{i-1}) + \frac{1}{2}(b_{i+\frac{1}{2}}^{in} + b_{i+\frac{1}{2}}^{out})(\gamma_i - \gamma_{i+1}) \\ = \delta_i \left(\frac{\gamma_{i+1} - \gamma_i}{h_i} - \frac{\gamma_i - \gamma_{i-1}}{h_{i-1}} \right) + \beta_{E,i} \left(\frac{\gamma_{i+1} - \gamma_{i-1}}{2} \right)^\perp. \end{aligned} \quad (3.29)$$

Po preusporiadaní tejto rovnice dostávame finálny tvar našej diskretizovanej semi-implicitnej schémy a dopad na diagonálnu dominanciu systému týchto rovníc má hlavne preusporiadanie členov b tak, že členy $b_{i-\frac{1}{2}}^{in}$ a $b_{i+\frac{1}{2}}^{in}$ ponecháme na ľavej „implicitnej“ strane rovnice, pretože dosahujú iba nezáporné hodnoty, a v pravej „explicitnej“ presunieme iba nekladné členy $b_{i-\frac{1}{2}}^{out}$ a $b_{i+\frac{1}{2}}^{out}$. Po preusporiadaní dostávame

$$\begin{aligned} - \left(\frac{\delta_i^n}{h_{i-1}^n} + \frac{b_{i-\frac{1}{2}}^{in,n}}{2} \right) \gamma_{i-1}^{n+1} + \left[\frac{h_{i-1}^n + h_i^n}{2\Delta t} + \delta_i^n \left(\frac{1}{h_i^n} + \frac{1}{h_{i-1}^n} \right) + \frac{b_{i-\frac{1}{2}}^{in,n}}{2} + \frac{b_{i+\frac{1}{2}}^{in,n}}{2} \right] \gamma_i^{n+1} - \left(\frac{\delta_i^n}{h_i^n} + \frac{b_{i+\frac{1}{2}}^{in,n}}{2} \right) \gamma_{i+1}^{n+1} \\ = \frac{h_{i-1}^n + h_i^n}{2\Delta t} \gamma_i^n + \beta_{E_i}^n \left(\frac{\gamma_{i+1}^n - \gamma_{i-1}^n}{2} \right)^\perp + \frac{1}{2} b_{i-\frac{1}{2}}^{out,n} (\gamma_i^n - \gamma_{i-1}^n) + \frac{1}{2} b_{i+\frac{1}{2}}^{out,n} (\gamma_i^n - \gamma_{i+1}^n) \end{aligned} \quad (3.30)$$

Znovu si zavedieme podobné označenia, ako v (3.17)

$$\begin{aligned} A_{i,IIOE}^n &= \frac{\delta_i^n}{h_{i-1}^n} + \frac{b_{i-\frac{1}{2}}^{in,n}}{2}, \\ B_{i,IIOE}^n &= \frac{h_{i-1}^n + h_i^n}{2\Delta t} + \delta_i^n \left(\frac{1}{h_i^n} + \frac{1}{h_{i-1}^n} \right) + \frac{b_{i-\frac{1}{2}}^{in,n}}{2} + \frac{b_{i+\frac{1}{2}}^{in,n}}{2}, \\ C_{i,IIOE}^n &= \frac{\delta_i^n}{h_i^n} + \frac{b_{i+\frac{1}{2}}^{in,n}}{2}, \\ D_{i,IIOE}^n &= \frac{h_{i-1}^n + h_i^n}{2\Delta t} \gamma_i^n + \beta_{E_i}^n \left(\frac{\gamma_{i+1}^n - \gamma_{i-1}^n}{2} \right)^\perp + \frac{1}{2} b_{i-\frac{1}{2}}^{out,n} (\gamma_i^n - \gamma_{i-1}^n) + \frac{1}{2} b_{i+\frac{1}{2}}^{out,n} (\gamma_i^n - \gamma_{i+1}^n). \end{aligned} \quad (3.31)$$

Potom naša sústava rovníc bude v tvare

$$-A_{i,IIOE}^n \gamma_{i-1}^{n+1} + B_{i,IIOE}^n \gamma_i^{n+1} - C_{i,IIOE}^n \gamma_{i+1}^{n+1} = D_{i,IIOE}^n. \quad (3.32)$$

Ukážme si, že systém takýchto rovníc je diagonálne dominantný pre ľubovoľne veľký časový krok Δt . Musí platiť vzťah

$$|B_{i,IIOE}^n| \geq |-A_{i,IIOE}^n| + |-C_{i,IIOE}^n|, \quad i = 1, \dots, m, \quad (3.33)$$

a po dosadení označení (3.31) dostávame túto nerovnicu v tvare

$$\left| \frac{h_{i-1}^n + h_i^n}{2\Delta t} + \delta_i^n \left(\frac{1}{h_i^n} + \frac{1}{h_{i-1}^n} \right) + \frac{b_{i-\frac{1}{2}}^{in,n}}{2} + \frac{b_{i+\frac{1}{2}}^{in,n}}{2} \right| \geq \left| \frac{\delta_i^n}{h_{i-1}^n} + \frac{b_{i-\frac{1}{2}}^{in,n}}{2} \right| + \left| \frac{\delta_i^n}{h_i^n} + \frac{b_{i+\frac{1}{2}}^{in,n}}{2} \right|. \quad (3.34)$$

Všetky členy, ktoré vystupujú v tejto rovnici dosahujú iba nezáporné hodnoty, tým pádom môžeme absolútne hodnoty odstrániť. Ďalej si všimnime, že druhý, tretí a štvrtý člen ľavej strany nerovnice, je v súčte rovný celej pravej strane nerovnice, tým pádom dostávame

nerovnicu

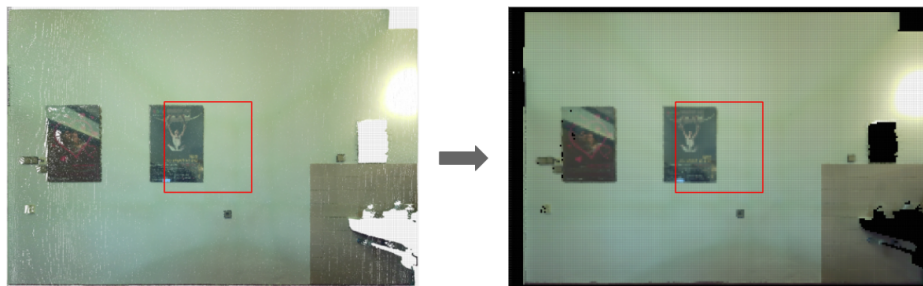
$$\frac{h_{i-1}^n + h_i^n}{2\Delta t} \geq 0, \quad (3.35)$$

ktorá platí pre ľubovoľne veľký kladný časový krok Δt .

Maticový zápis tohoto systému vyzerá podobne ako (3.19), len namiesto A_i^n máme $A_{i,IIOE}^n$ a podobne.

3.2 Sieť rovinného mračna bodov

Vzhľadom na to, že chceme, aby pohyb každého bodu krivky bol ovplyvnený vlastnosťami okolitých bodov mračna, by bolo časovo náročné počítať, ktoré body mračna sú v blízkosti bodov krivky, a aké sú v nich hodnoty. Preto si reprezentáciu mračna zjednodušíme, a to tak, že preložíme rovnomernú štvorcovú sieť cez mračno a budeme počítať priemernú hodnotu atribútov z bodov mračna, ktoré spadajú do jednotlivých štvorcov. Ako spomíname v kapitole 1, k dispozícii máme mračno bodov s niekoľkými atribútmi ako napr. farba, intenzita, odchýlka normál a po segmentácii mračna bodov do rovín vieme použiť aj kolmú vzdialenosť od regresnej roviny. Farbu máme reprezentovanú pomocou troch kanálov R,G a B, pričom každý z týchto kanálov analyzujeme zvlášť. Na sieti budeme počítať priemernú hodnotu všetkých týchto atribútov rovnakým spôsobom, graficky si to znázorníme na farbe a pre lepšiu predstavu si ukážeme aj vizuálne porovnanie pixelovej reprezentácie a reprezentácie mračnom bodov. Na obrázku 3.3 je vizualizovaná jedna z vysegmentovaných stien, pričom vľavo mô-



Obr. 3.3: Porovnanie reprezentácie mračnom bodov a pixelovej reprezentácie.

žeme vidieť reprezentáciu mračnom bodov a na pravej strane pixelovú reprezentáciu. V oboch obrázkoch sú červené obdĺžniky, ktoré znázorňujú detailnejší výsek, ktorý vidíme na obrázku 3.4.



Obr. 3.4: Porovnanie reprezentácie mračnom bodov a pixelovej reprezentácie v detailnom zábere.

V tomto detailnejšom zábere môžeme lepšie vidieť body mračna v porovnaní s pixelovou reprezentáciou. Pre ešte lepšiu predstavu si ukážeme na obrázku 3.5 ešte detailnejší výrez z obrázku 3.4. V tomto výreze je presne vidieť, ktoré body spadajú do jednotlivých štvorcov



Obr. 3.5: Porovnanie reprezentácie mračnom bodov a pixelovej reprezentácie v detailnom zábere.

siete, a priemer ich hodnôt na štvorcoch môžeme vidieť na pravo vo vizualizácii pixelovej reprezentácie. Pixelovú reprezentáciu budeme ďalej označovať I a dĺžku hrany pixela budeme označovať h .

3.2.1 Odchýlka normál

V predchádzajúcich sekciách hovoríme o vlastnosti mračna, ktorú voláme odchýlka normál. Túto vlastnosť počítame pre každý bod mračna na základe polohy bodov a intuitívne si ju môžeme predstaviť, ako mieru krivosti mračna. Pre každý bod nájdeme K najbližších bodov mračna a preložíme cez ne regresnú rovinu. Ďalej porovnávame normálové vektory regresných rovín okolitých K bodov nasledujúcim vzťahom

$$\Delta N_i = 1 - \frac{1}{K} \sum_{j=1}^K |N_i \cdot N_j|. \quad (3.36)$$

Takýmto spôsobom vieme odhaliť miesta, kde je väčšia krivosť, ako napríklad hrany obrazov, vypínačov, zárubní a podobne.

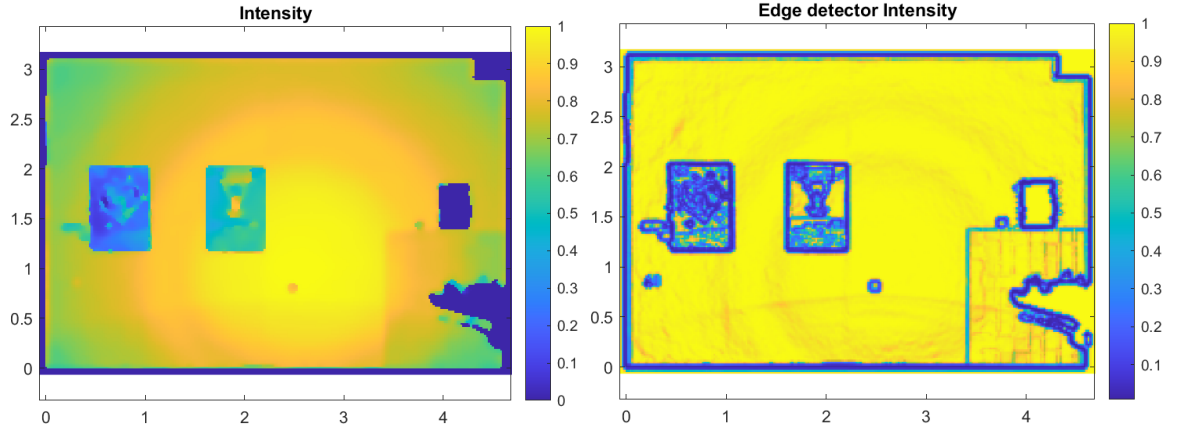
3.2.2 Výpočet hranového detektora na pixelovej reprezentácii vlastností rovinného mračna

Hranový detektor využívame v rovnicach pre výpočet normálovej rýchlosti. Jeho výpočet popisujú rovnice (2.25), (2.26) a (2.27). V týchto rovnicach využívame hodnoty hranového detektora v miestach, kde leží krivka. V praktickej časti si hranový detektor vypočítame pred tým, ako začneme vyvíjať krivku a v priebehu vývoja krivky berieme do úvahy hodnoty hranového detektora v pixeli, v ktorom leží daný bod krivky, pre ktorý počítame vývoj. Rovnica (2.27) bude po diskretizácii vyzeráť nasledovne

$$s_p = \|\nabla(G_{\sigma_0} * I)_p\|, \quad (3.37)$$

pričom s_p je hodnota s v pixeli p . Potom I z tejto rovnice bude v diskretnom prípade naša pixelová reprezentácia mračna, na ktorú aplikujeme konvolúciu s Gaussovským konvolučným jadrom a následne počítame gradienty aproximáciou konečnými diferenciami. Používame centrálnu diferenciu v prípade, že element neleží na hranici oblasti a spätnú alebo doprednú, ak element leží na hranici. Z gradientov nakoniec vypočítame normy. Ďalej už len aplikujeme rovnicu (2.26) na výpočet hranového detektora $g(s_p, k)$, ktorý následne zhladíme použitím vzťahu (2.25). Tento proces aplikujeme na vlastnosti, ktoré máme k dispozícii, a tak dostaneme hranové detektory zvlášť pre všetky tieto vlastnosti. Tentokrát si to ukážeme na intenzite a použijeme rovnakú stenu z obrázka 3.3, na ktorej sme si ukázali prechod z reprezentácie mračnom bodov na pixelovú reprezentáciu.

Na ľavej strane obrázka 3.6 môžeme vidieť intenzitu odrazených lúčov a na pravej strane hranový detektor vypočítaný z intenzity. Podľa farebnej škály môžeme vidieť, že na miestach,



Obr. 3.6: Intenzita a hranový detektor vypočítaný z intenzity.

kde je hrana objektov, dosahuje hranový detektor hodnoty blízke 0, zatiaľ čo na miestach, kde nie je žiadny skok intenzity, sú hodnoty blízke 1. Po tom, čo sú vypočítané týmto spôsobom všetky hranové detektory, sú všetky spojené do jedného výsledného hranového detektora, s ktorým budeme pracovať v rovnicach pre vývoj krivky. Spôsob, akým budeme tieto hranové detektory spájať, popisujeme v sekcii 3.5.

3.3 Diskretizácia tangenciálnej rýchlosti

Na tangenciálnu rýchlosť používame rovnicu (2.15), ktorú budeme v tejto podkapitole diskretizovať. Ľavú stranu rovnice (2.15): $\partial_s \alpha$ si nahradíme centrálnou diferenciou, čiže v strede hrany h_i^n v bode $\gamma_{i+\frac{1}{2}}$

$$\partial_s \alpha = \frac{\alpha_{i+1}^n - \alpha_i^n}{h_i^n}. \quad (3.38)$$

Členy na pravej strane rovnice (2.15) si vyjadríme v bode $\gamma_{i+\frac{1}{2}}^n$. Ako budeme aproximovať funkciu κ^\pm , si ukážeme trošku neskôr. Funkciu β budeme v bode $\gamma_{i+\frac{1}{2}}^n$ aproximovať pomocou priemeru

$$\beta_{i+\frac{1}{2}}^n = \frac{1}{2}(\beta_i^n + \beta_{i+1}^n), \quad (3.39)$$

príčom aproximáciu hodnôt β_i^n a β_{i+1}^n budeme detailne rozvádzať v ďalšej podkapitole 3.4. Integrál v rovnici (2.15) budeme aproximovať pomocou Riemannovej sumy ako

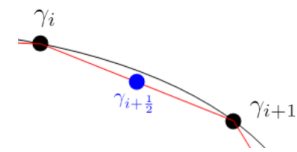
$$\int_{\text{Im} \gamma} \kappa^\pm \beta ds \approx \sum_{i=1}^m \kappa_{i+\frac{1}{2}}^{\pm n} \beta_{i+\frac{1}{2}}^n h_i^n. \quad (3.40)$$

Dĺžku krivky $L(\gamma^t)$ budeme aproximovať súčtom dĺžok všetkých hrán krivky

$$L(\gamma^t) \approx L^n = \sum_{i=1}^m h_i, \quad (3.41)$$

Normu $\|\partial_u \gamma\|$ si numericky aproximujeme nasledovne

$$\|\partial_u \gamma\| \approx \frac{\|\gamma_{i+1}^n - \gamma_i^n\|}{\Delta u}, \quad (3.42)$$



pričom pravú stranu s využitím (3.2) a (3.8) môžeme prepísať na tvar

$$\frac{\|\gamma_{i+1}^n - \gamma_i^n\|}{\Delta u} = h_i^n m. \quad (3.43)$$

Aplikovaním všetkých týchto aproximácií, dostávame rovnicu (2.15) v diskretizovanom tvare

$$\frac{\alpha_{i+1}^n - \alpha_i^n}{h_i^n} = \kappa_{i+\frac{1}{2}}^{\pm n} \beta_{i+\frac{1}{2}}^n - \frac{1}{L^n} \sum_{j=1}^m \kappa_{j+\frac{1}{2}}^{\pm n} \beta_{j+\frac{1}{2}}^n h_j^n + \left(\frac{L^n}{h_i^n} - 1 \right) \omega. \quad (3.44)$$

Túto rovnicu si ešte upravíme tak, aby na ľavej strane bol len člen α_{i+1}

$$\alpha_{i+1}^n = \alpha_i^n + h_i^n \left(\kappa_{i+\frac{1}{2}}^{\pm n} \beta_{i+\frac{1}{2}}^n - \frac{1}{L^n} \sum_{j=1}^m \kappa_{j+\frac{1}{2}}^{\pm n} \beta_{j+\frac{1}{2}}^n h_j^n + \left(\frac{L^n}{h_i^n} - 1 \right) \omega \right). \quad (3.45)$$

Aby sme vedeli použiť túto rovnicu, tak si musíme v jednom z bodov krivky určiť počiatočnú hodnotu. V našom prípade si zadáme $\alpha_1^n = 0$, a na základe toho vypočítame všetky ďalšie α_i^n , $i = 2, \dots, m$. Po výpočte týchto hodnôt vypočítame ich priemer $\bar{\alpha}^n$ a od všetkých α_i^n ho odčítame, aby výsledný priemer hodnôt bol rovný 0.

$$\alpha_i^n := \alpha_i^n - \bar{\alpha}^n, \quad i = 1, \dots, m.$$

Diskretizácia krivosti

Pri odvodzovaní diskretizovanej rovnice pre tangenciálnu rýchlosť, sme použili aproximáciu funkcie znamienkovej krivosti v bode $\gamma_{i+\frac{1}{2}}^n$, ale nevysvetlili sme si, ako ju počítame. Znamienkovú krivosť κ^{\pm} analyticky počítame ako

$$\kappa^{\pm} = \frac{a_N}{\|\partial_u \gamma\|^2}. \quad (3.46)$$

Člen a_N je zrýchlenie v zmysle kráčania po krivke vo fixovanom čase v smere pozitívne orientovanej normály N^+ , preto ho vieme napísať ako $a_N = a \cdot N^+$, kde $a = \partial_u^2 \gamma$. V menovateli vystupuje člen $\|\partial_u \gamma\|$, ktorý označuje rýchlosť v zmysle kráčania po krivke. Rovnicu (3.46) potom vieme zapísať takto

$$\kappa^{\pm} = \frac{\partial_u^2 \gamma \cdot N^+}{\|\partial_u \gamma\|^2}. \quad (3.47)$$

Druhú deriváciu aproximujeme centrálnou diferenciou a N^+ v bode $\gamma_{i+\frac{1}{2}}$

$$\partial_u^2 \gamma \approx \frac{\gamma_{i+\frac{3}{2}} - 2\gamma_{i+\frac{1}{2}} + \gamma_{i-\frac{1}{2}}}{\Delta u^2}. \quad (3.48)$$

Ďalej na aproximáciu menovateľa v rovnici (3.47), použijeme rovnicu (3.42), a dostávame

$$\kappa_{i+\frac{1}{2}}^{\pm} = \frac{\frac{\gamma_{i+\frac{3}{2}} - 2\gamma_{i+\frac{1}{2}} + \gamma_{i-\frac{1}{2}}}{\Delta u^2} \cdot N_{i+\frac{1}{2}}^+}{\frac{\|\gamma_{i+1}^n - \gamma_i^n\|^2}{\Delta u^2}}. \quad (3.49)$$

Ďalej použijeme na úpravu členov v čitateli $\gamma_{i-\frac{1}{2}}$ a $\gamma_{i+\frac{1}{2}}$ vzťahy (3.5), (3.6) a analogicky pre $\gamma_{i+\frac{3}{2}}$. Na úpravu menovateľa použijeme rovnicu (3.8) a dostávame

$$\kappa_{i+\frac{1}{2}}^{\pm} = \frac{\frac{\gamma_{i+1} + \gamma_{i+2}}{2} - (\gamma_i + \gamma_{i+1}) + \frac{\gamma_{i-1} + \gamma_i}{2}}{h_i^2} \cdot N_{i+\frac{1}{2}}^+. \quad (3.50)$$

Kladne orientovaný normálový vektor $N_{i+\frac{1}{2}}^+$ si vyjadríme vzťahom

$$N_{i+\frac{1}{2}}^+ = \begin{pmatrix} r_i \\ h_i \end{pmatrix}^{\perp}, \quad (3.51)$$

kde r_i je vektor z bodu γ_i do bodu γ_{i+1} , a preto ho vieme zapísať nasledovne

$$r_i = \gamma_{i+1} - \gamma_i. \quad (3.52)$$

Aplikovaním tohoto vzťahu a menšími úpravami dostávame diskretizáciu krivosti v tvare

$$\kappa_{i+\frac{1}{2}}^{\pm} = \frac{\gamma_{i+2} - \gamma_{i+1} - \gamma_i + \gamma_{i-1}}{2h_i^3} \cdot (\gamma_{i+1} - \gamma_i)^{\perp}. \quad (3.53)$$

3.4 Diskretizácia normálovej rýchlosti

Pre normálovú rýchlosť používame rovnicu (2.16). Túto rovnicu budeme diskretizovať v bodoch krivky $\gamma_i, i = 1, \dots, m$, a pre i -ty bod dostaneme rovnicu

$$\beta_i = \delta_i \kappa_i^{\pm} + \beta_{E,i}. \quad (3.54)$$

Akým spôsobom aproximujeme jednotlivé funkcie z rovnice (3.54) popisujeme v sekciách 3.4.1 a 3.4.2

3.4.1 Diskretizácia krivostnej regularizácie

Prvá časť rovnice (2.16): $\delta_i \kappa_i^{\pm}$ je označenie krivostnej regularizácie v našom diskretnom matematickom modeli. V predchádzajúcej kapitole sme si odvodili, ako numericky aproximujeme funkciu krivosti v bode $\gamma_{i+\frac{1}{2}}$, ale v tejto časti chceme krivosť aproximovať v bode γ_i . Preto ju budeme počítat', ako priemer krivostí v bodoch $\gamma_{i+\frac{1}{2}}$ a $\gamma_{i-\frac{1}{2}}$

$$\kappa_i^{\pm} = \frac{\kappa_{i+\frac{1}{2}}^{\pm} + \kappa_{i-\frac{1}{2}}^{\pm}}{2}. \quad (3.55)$$

Funkciu δ_i popisuje vzťah (2.17). Funkciu $\lambda(t)$ budeme v diskretnom prípade označovať λ_n , a ako spomíname v sekcii 2.3.2, tento parameter nastavujeme na začiatku časového vývoja na hodnotu 0.5 a jeho hodnotu meníme, keď je vývoj krivky skoro pri konci na hodnotu 1. Ďalej pre člen $\delta_{\text{local}}(u, t)$ platí v analytickom vyjadrení rovnica (2.18). V diskretizovanom prípade to znamená, že berieme do úvahy hodnotu $\beta_{E,i}$. O tom, ako presne budeme diskretizovať $\beta_{E,i}$, budeme hovoriť v podkapitole 3.4.2. Hranový detektor $g_1(\gamma(u, t))$ v bode krivky budeme aproximovať nasledovne

$$g_1(\gamma(u, t)) \approx g_1(\gamma_i), \quad (3.56)$$

pričom $g_1(\gamma_i)$ je hodnota hranového detektora, v pixeli, v ktorom leží bod γ_i . Po aplikácii týchto aproximácií dostávame diskretizovanú rovnicu pre krivostnú regularizáciu v tvare

$$\delta_i = (1 - \lambda_n)(\delta_{\text{global}} \cdot \delta_{\text{local},i} \cdot g_1(\gamma_i)). \quad (3.57)$$

3.4.2 Diskretizácia normálovej rýchlosti riadenej vonkajším vplyvom

Člen $\beta_{E,i}$ z rovnice (3.54) získame diskretizáciou rovnice (2.21) v bode γ_i . V predchádzajúcej podkapitole 3.4.1 sme si vysvetlili, ako aproximujeme funkcie $\lambda(t)$ a $g_1(\gamma(u, t))$. Ďalej si ukážeme, ako budeme aproximovať N^+ . Pre N^+ platí vzťah (2.32), ktorý budeme aproximovať použitím centrálnej diferencie v bode γ_i nasledovne

$$N^+ \approx N_i^+ = \left(\frac{\gamma_{i+1} - \gamma_{i-1}}{\|\gamma_{i+1} - \gamma_{i-1}\|} \right)^{\perp}. \quad (3.58)$$

Diskretizovaná rovnica (2.21) bude v tvare

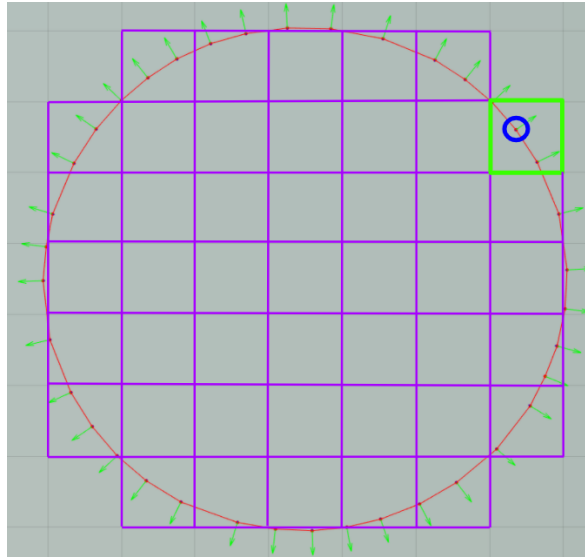
$$\beta_{E,i} = (1 - \lambda_n)H(\gamma_i) \cdot g_1(\gamma_i) + \lambda_n((-\nabla g_1(\gamma_i)) \cdot N_i^+). \quad (3.59)$$

Gradient $\nabla g_1(\gamma_i)$ budeme vo vnútri oblasti aproximovať centrálnymi diferenciami a na hraničiach oblasti použijeme spätné a dopredné diferencie. Celý druhý člen tejto rovnice slúži na

príťahovanie krivky k hranám, ktoré boli nájdené hranovým detektorom. Ďalej si ukážeme, ako budeme počítat' funkciu homogenity $H(\gamma_i)$.

Výpočet funkcie homogenity

Funkciu homogenity budeme v diskretnom prípade značiť $H^l(\gamma_i)$, pričom horný index l značí l -tú vlastnosť mračna (R, G, B, intenzita,...). Funkciu homogenity počítame separátne pre všetky vlastnosti a po výpočte ich agregujeme do jednej výslednej funkcie homogenity podľa postupu v sekcii 3.5.3. Ukážeme si, ako počítame funkciu homogenity pre jednu z vlastností. Pre i -ty bod krivky γ nás zaujíma odlišnosť vlastnosti v pixeli, v ktorom tento bod krivky leží, od priemernej hodnoty vlastnosti v pixeloch prejdejších krivkou od začiatku vývoja, množinu týchto pixelov označujeme Ω^n . Pre lepšie pochopenie si ukážeme vizualizáciu na obrázku 3.7. Bod krivky γ_i je na obrázku zobrazený v modrej kružnici. Pixel, v ktorom



Obr. 3.7: Vizualizácia pixelov vo vnútri krivky a pixelu, v ktorom leží bod γ_i .

tento bod leží, je označený zeleným štvorcikom a oblasť Ω^n je vyznačená fialovými štvorcami. Priemernú hodnotu na oblasti Ω^n budeme počítat' ako

$$\overline{I_{\Omega^n}^l} = \frac{1}{|\Omega^n|} \sum_{p \in \Omega^n} I^l(p)h^2. \quad (3.60)$$

Potom odlišnosť zeleného pixela od pixelov množiny Ω^n budeme počítat' nasledovne

$$d_i^n = \left| I^l(\gamma_i^n) - \overline{I_{\Omega^n}^l} \right| \quad (3.61)$$

Diskretizovaná funkcia homogenity potom vyzerá nasledovne

$$H_i^{l,n} = \begin{cases} 1 - \frac{d_i^n}{\mu^l}, & d_i^n \leq \mu^l \\ 0, & \text{inak} \end{cases}. \quad (3.62)$$

Čím menší je rozdiel d_i , tým väčšia bude výstupná hodnota z tejto funkcie, a tým rýchlejšie sa bude bod γ_i pohybovať.

Naším cieľom je použiť všetky vlastnosti, ktoré máme k dispozícii, preto budeme funkciu homogenity počítat' separátne pre všetky vlastnosti a výsledné funkcie homogenity budeme spájať. O tom, akým spôsobom ich budeme spájať, budeme hovoriť v podkapitole 3.5.3.

3.5 Spájanie hranových detektorov a funkcií homogenity

V tejto podkapitole si povieme viac o spájaní hranových detektorov a funkcií homogenity. Tieto funkcie sa počítajú pomocou vlastností mračna bodov (farba, intenzita, odchýlka normál, vzdialenosť od regresnej roviny) a budeme ich spájať pomocou Fuzzy logiky.

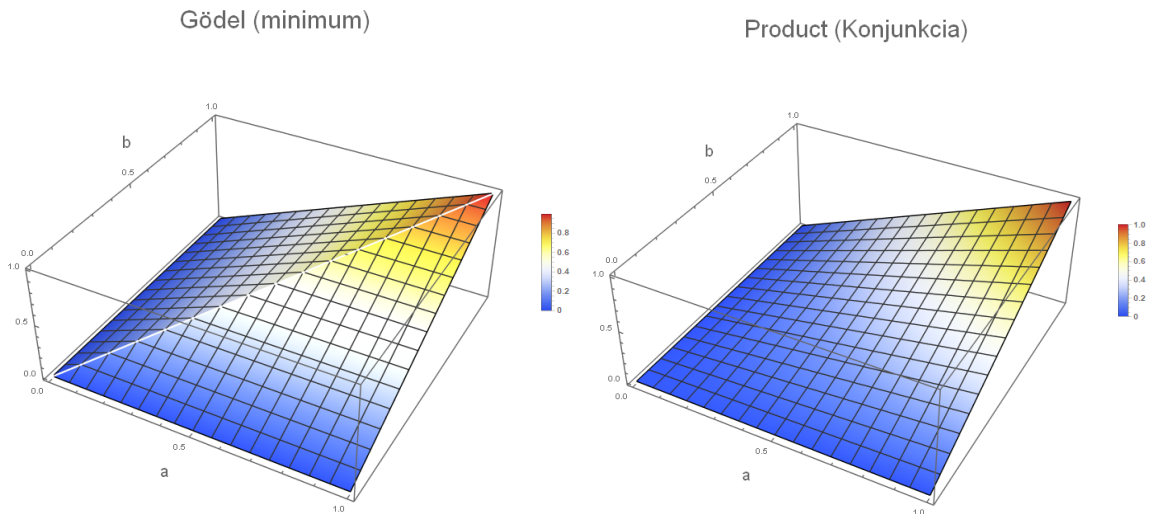
3.5.1 Fuzzy logika

Informácie o fuzzy logike sme čerpali z článku [6]. Rozdiel medzi fuzzy logikou a klasickou logikou je, že klasická logika uvažuje iba hodnoty 1 a 0 pre pravdu a nepravdu, zatiaľ čo fuzzy logika uvažuje reálne čísla na intervale $[0, 1]$. Nás budú zaujímať hlavne fuzzy operátory \wedge (konjunkcia) a \vee (disjunkcia) a \neg (negácia). Rovnako, ako pri klasickej logike, tak aj vo fuzzy logike znamená 0 úplnú nepravdu a 1 úplnú pravdu. Majme dve hodnoty a a $b \in [0, 1]$. Fuzzy analógia konjunkcie $a \wedge b$ sa počíta pomocou tzv. T-noríem, ktoré definujeme ako zobrazenia $T : [0, 1]^2 \rightarrow [0, 1]$. Existuje niekoľko prístupov, ako počítať hodnotu pre fuzzy konjunkciu dvoch premenných. V našej práci pracujeme s dvoma prístupmi na výpočet T-noríem, ktoré uvádzame v tabuľke 3.1. Pre lepšie pochopenie si ukážeme vizualizáciu týchto dvoch typov

Tabuľka 3.1: T-normy.

Meno	T-norma
Gödel (minimum)	$T_G(a, b) = \min(a, b)$
Súčinový	$T_P(a, b) = a \cdot b$

T-noríem na obrázku 3.8.



Obr. 3.8: Porovnanie Fuzzy T-noríem.

Fuzzy negácia je funkcia $N : [0, 1] \rightarrow [0, 1]$, kde $N(0) = 1$ a $N(1) = 0$. My budeme používať negáciu, pre ktorú platí nasledovný vzťah

$$N(a) = 1 - a. \quad (3.63)$$

Fuzzy disjunkciu budeme počítať pomocou T-konoríem, ktoré definujeme ako funkciu $S : [0, 1]^2 \rightarrow [0, 1]$ a ich tvar sa dá ľahko odvodiť z T-noríem pomocou De Morganovho pravidla klasickej logiky

$$a \vee b = N(N(a) \wedge N(b)), \quad (3.64)$$

S použitím negácie (3.63) T-konormu budeme počítať takto

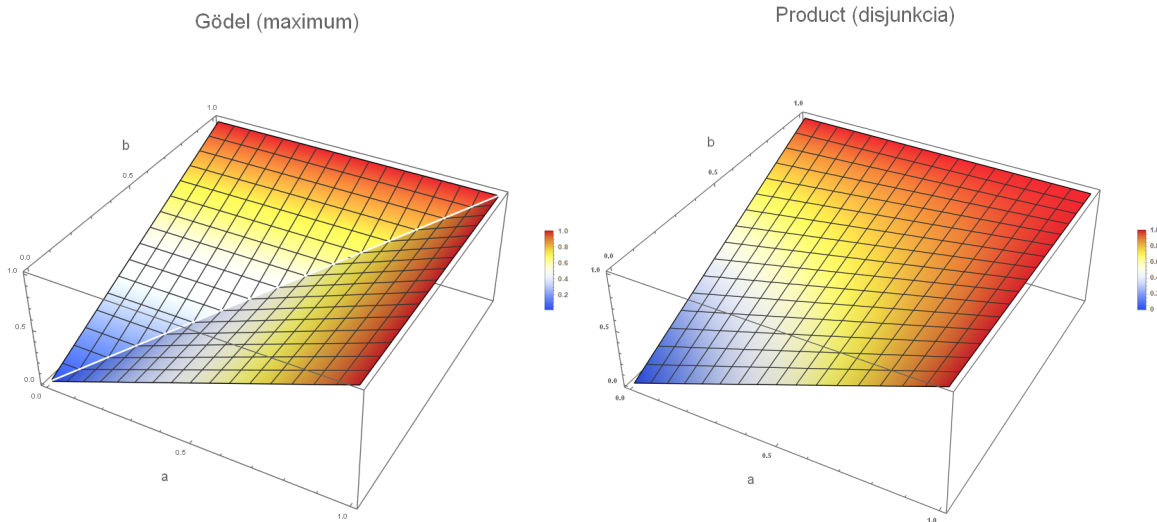
$$S(a, b) = 1 - T(1 - a, 1 - b). \quad (3.65)$$

Pre T-normy uvedené v tabuľke 3.1, dostávame použitím vzťahu (3.65) vzťahy pre T-konormy uvedené v tabuľke 3.2.

Tabuľka 3.2: T-konormy.

Meno	T-konorma
Gödel (maximum)	$S_G(a, b) = \max(a, b)$
Súčinový	$S_P(a, b) = a + b - a \cdot b$

Vizualizáciu T-konorm vidíme na obrázku 3.9. Gödelovu T-normu a T-konormu budeme



Obr. 3.9: Porovnanie Fuzzy T-konorm.

nazývať aj MinMaxová T-norma resp. T-konorma.

Teraz, keď máme zadané fuzzy operátory pre dva vstupné parametre, tak si ukážeme, ako vieme agregovať n vstupných parametrov. Operátor agregácie vieme použitím T-norm zapísať takto

$$\begin{aligned} A_T(a) &= a, \\ A_T(a_1, \dots, a_n) &= T(a_1, A_T(a_2, \dots, a_n)). \end{aligned} \quad (3.66)$$

Analogicky to platí pre T-konormy. Použitím vzťahov z tabuliek 3.1 a 3.2 dostávame vzťahy pre agregačné operátory uvedené v tabuľke 3.3.

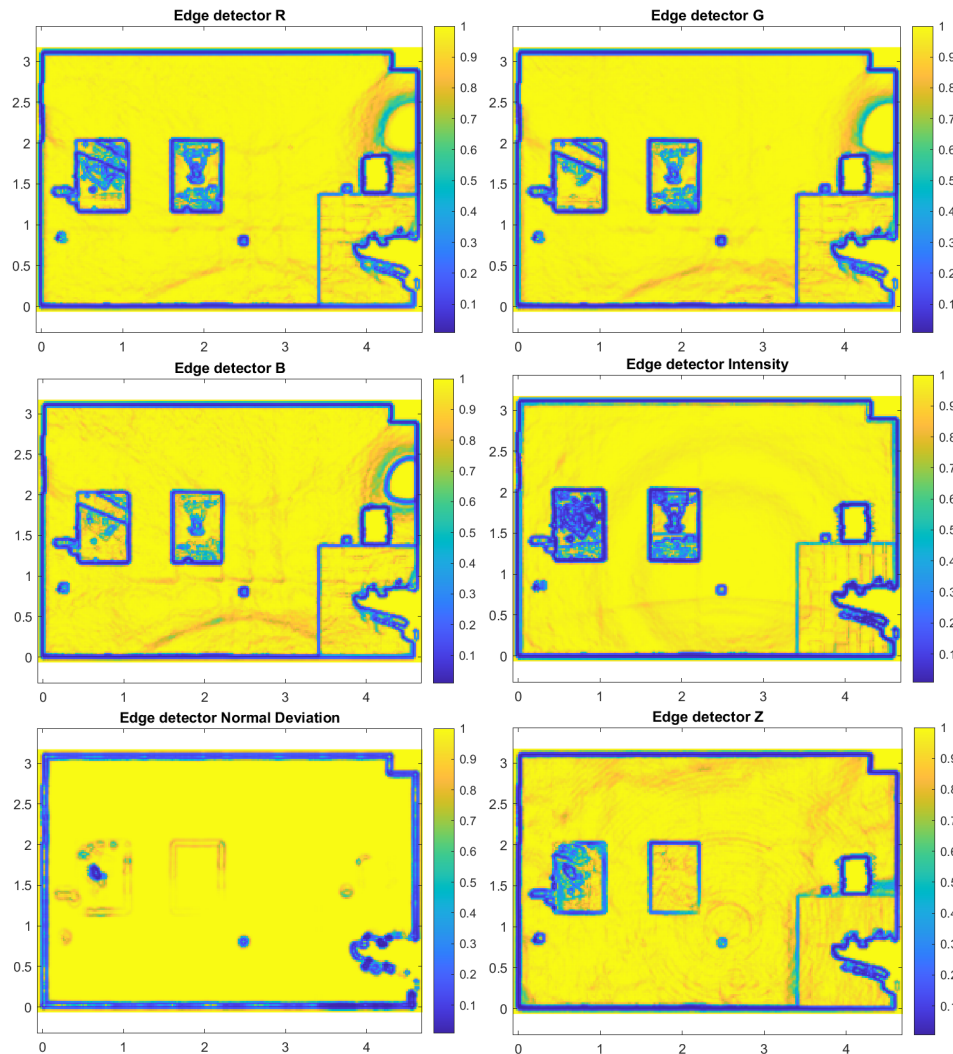
Tabuľka 3.3: Agregačné operátory pre T-normy a T-konormy.

Meno	T-normy	Agregovaný operátor
Gödel (minimum)	T_G	$A_{T_G}(a_1, \dots, a_n) = \min(a_1, \dots, a_n)$
Súčinový (konjunkcia)	T_P	$A_{T_P}(a_1, \dots, a_n) = \prod_{i=1}^n a_i$
Gödel (maximum)	S_G	$A_{S_G}(a_1, \dots, a_n) = \max(a_1, \dots, a_n)$
Súčinový (disjunkcia)	S_P	$A_{S_P}(a_1, \dots, a_n) = 1 - \prod_{i=1}^n (1 - a_i)$

3.5.2 Spájanie hranových detektorov

Hranové detektory počítame separátne pre 6 vlastností: 3 zložky farby, ktoré budeme označovať R, G a B, intenzitu, lokálnu odchýlku normál a vzdialenosť od regresnej roviny. Na obrázku 3.10 môžeme vidieť, ako vyzerajú tieto hranové detektory separátne pre stenu, ktorej farbu vizualizovanú spojením kanálov R, G a B môžeme vidieť na obrázku 3.3, a intenzitu

na obrázku 3.6. Vizualizáciu pixelových reprezentácií všetkých vlastností tejto steny môžeme vidieť na obrázku 4.11.



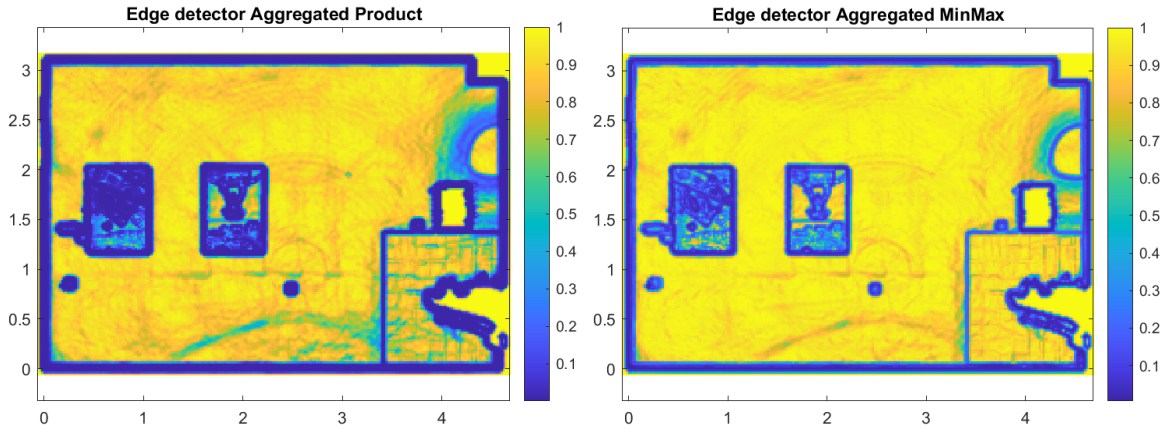
Obr. 3.10: Hranové detektory separátne pre všetky vlastnosti.

Na hranových detektoroch pre farebné kanály R, G a B môžeme pozorovať v pravom hornom rohu časť, kde bola nájdená hrana, pričom na ostatných hranových detektoroch sa táto hrana nenašla. To je spôsobené tým, že na túto časť steny svietila lampa a to spôsobilo, že sa v zložkách farby našla táto hrana a v ostatných vlastnostiach nie. Hranové detektory budeme agregovať T-normami, ktoré predstavujú fuzzy konjunkciu a teda, aby bol pixel označený hodnotou blízkou 1, čo značí, že pixel neprislúcha hrane, tak tento pixel nemôže prislúchať hrane v žiadnom z hranových detektorov. Všetky tieto hranové detektory sú vstupné parametre do agregáčného operátora A_T . Agregovaný hranový detektor dostaneme ako

$$g_1 = A_T(g_1^1, \dots, g_1^6) \quad (3.67)$$

a výsledné hranové detektory pre agregáčné operátory A_{T_G} a A_{T_P} vidíme na obrázku 3.11.

Takto získame dva možné agregované detektory a v našom programe vieme nastaviť, ktorý z nich sa bude využívať vo výpočtoch normálovej rýchlosti. V agregovaných hranových detektoroch na obrázku 3.11 môžeme vidieť, že hrana v pravom dolnom rohu, ktorá oddeľuje maľovanú stenu od obkladu, je oveľa výraznejšia v súčinovom (produktovom) hranovom detektore, ako v MinMaxovom. Dôvodom je, že táto hrana bola nájdená v takmer každom hranovom detektore a súčinom hodnôt blízkyh 0, ktoré sa vyskytujú na hranách sa táto hrana zvýraznila na rozdiel od MinMaxového agregátora, ktorého výstupom je minimálna hodnota



Obr. 3.11: Hranové detektory separátne pre všetky vlastnosti.

nájdená vo vstupných hranových detektoroch. Agregovaním všetkých hranových detektorov pomocou súčinového operátora môžeme odhaliť, ktoré hrany sú relevantnejšie z hľadiska výskytu vo viacerých hranových detektoroch, keďže prenásobením hodnôt blízkych 0 sa výsledná hodnota ešte zmenší, čo znamená, že hrany budú ešte výraznejšie.

3.5.3 Spájanie funkcií homogenity

Podobne, ako pre hranové detektory, spájame v tomto prípade všetkých 6 vlastností, na výpočet funkcie homogenity H . V sekcii 3.4.2 sme si ukázali, akým spôsobom vypočítame funkciu homogenity pre jednu z vlastností. Výslednú funkciu homogenity budeme počítať použitím agregáčnych operátorov A_{S_G} alebo A_{S_P} z tabuľky 3.3, pričom vstupné parametre budú jednotlivé funkcie homogenity

$$H_i^n = A_S(H_i^{1,n}, \dots, H_i^{1,n}). \quad (3.68)$$

Na agregovanie funkcií homogenity používame T-konormové agregátory, ktoré vo fuzzy logike predstavujú disjunkciu. V tomto prípade to znamená, že stačí, aby funkcia homogenity aspoň podľa jednej z vlastností dosahovala vyššie hodnoty, a výsledná funkcia homogenity bude dosahovať tiež vyššie hodnoty.

3.6 Topologické zmeny

Počas vývoja kriviek sa stáva, že do seba krivky „narazia“. V prípade, že sa k sebe priblížia dve rôzne krivky, chceme, aby sa spojili do jednej. Ďalšou možnosťou je, že sa k sebe priblížia body rovnakej krivky, napríklad, keď krivka obíde nejaký objekt ležiaci v stene. Vtedy chceme rozdeliť krivku na dve, preto používame algoritmy na spájanie a rozdeľovanie kriviek. Algoritmus, ktorý používame je popísaný v článkoch [1] a [2]. Bežné algoritmy na riešenie topologických zmien majú výpočtovú náročnosť $O(n^2)$, kde n je počet bodov krivky. Tieto algoritmy sú založené na počítaní vzájomných vzdialeností všetkých bodov krivky. Algoritmus, ktorý používame my, má výpočtovú náročnosť $O(n)$, vďaka čomu je rozdeľovanie a spájanie kriviek veľmi rýchle. Tento algoritmus sa skladá z troch častí. Prvá časť je odoberanie a pridávanie bodov krivky, druhá je rozdeľovanie kriviek a tretia spájanie kriviek.

3.6.1 Pridávanie a odoberanie bodov

Úlohou prvej časti algoritmu je pridávať body na miestach, kde sa body krivky vzdalujú a odoberať body na miestach, kde sú body krivky príliš nahusto. Kvôli ďalším častiam algoritmu

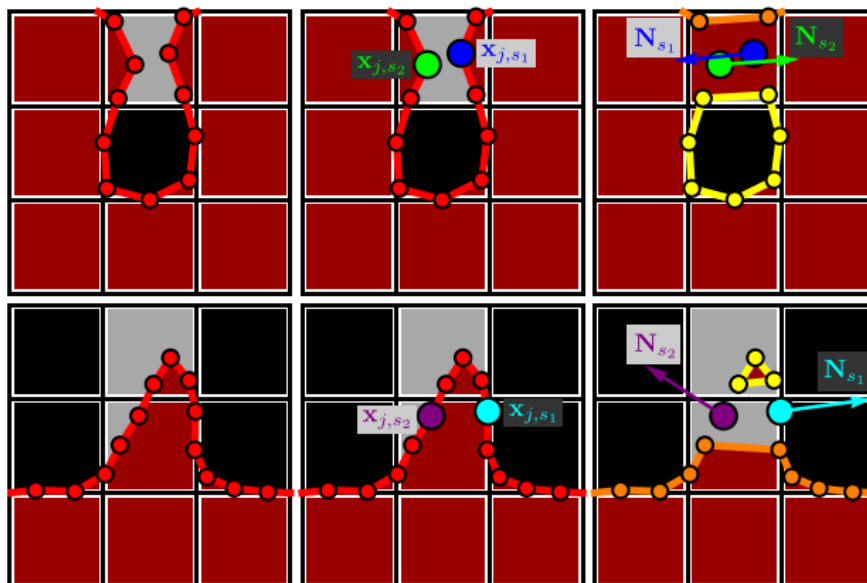
je dôležité udržiavať susedné body približne v rovnakej vzdialenosti h_d , ktorá je rovná polovici dĺžky hrany pixela $h_d = \frac{h}{2}$.

Túto časť algoritmu rozdelíme do niekoľkých krokov:

1. Prechádzame postupne po hranách krivky a pripočítavame rozdiely dĺžky hrany h_i a želanej dĺžky hrany h_d do premennej $L_{dif} = h_i - h_d$.
2. Po každom pripočítaní do premennej L_{dif} skontrolujeme, či je jej absolútna hodnota väčšia ako hodnota h_d , ak je väčšia pokračujeme krokom 3.
3. V prípade, že je absolútna hodnota $|L_{dif}|$ väčšia, ako h_d rozlišujeme dve situácie:
 - (a) Ak je $L_{dif} > 0$, tak chceme pridávať body, preto si nájdeme najdlhšiu hranu h_{max} na úseku, na ktorom sme počítali rozdiely h_i a h_d . Na tejto hrane pridáme N bodov, pričom $N = \lfloor \frac{h_{max}}{h_d} \rfloor$ ¹. Následne prepočítame $L_{dif} = L_{dif} - N \cdot h_d$.
 - (b) Ak je $L_{dif} < 0$, to znamená, že na danom úseku, na ktorom sme túto hodnotu počítali, sú body príliš nahusto, preto z neho budeme odoberať jeden bod. Odstránime taký bod γ_i , ktorého súčet vzdialeností $h_{i-1} + h_i$ je najmenší na danom úseku. Nakoniec prepočítame $L_{dif} = L_{dif} + h_d$.
4. Na konci algoritmu skontrolujeme, či má krivka viac, ako minimálny požadovaný počet bodov. Ak má menej bodov, tak ju vymažeme.

3.6.2 Rozdeľovanie krivky

Krivku chceme rozdeliť napríklad v prípade, že obide objekt ležiaci v stene, a jej body sa k sebe približujú. Na to, aby sme krivku rozdelili, musí byť splnených niekoľko podmienok. Označme indexy bodov, v ktorých chceme krivku rozdeliť s_1 a s_2 . Jednou z podmienok je, aby absolútna hodnota rozdielu indexov s_1 a s_2 , bola väčšia ako 3. Ďalšou podmienkou, je aby normály v týchto bodoch, smerovali opačným smerom, a zároveň, aby normála N_{s_1} smerovala k bodu s_2 a naopak. Pre lepšie pochopenie si to ukážeme na obrázku 3.12.



Obr. 3.12: Ukážka prípadu, kedy chceme, aby sa krivka rozdelila (obrázky v prvom riadku) a prípadu kedy nechceme, aby sa krivka rozdelila (obrázky v druhom riadku). Tento obrázok pochádza z článku [1].

¹Zátvorky $\lfloor \]$ označujú zaokrúhlenie nadol

Na obrázkoch v prvom riadku môžeme vidieť, že krivka obklopí pixel a sú splnené obe podmienky, normály smerujú opačným smerom a aj indexová vzdialenosť bodov je väčšia, ako 3. Na obrázkoch v druhom riadku vidíme, že krivka sa snaží vojsť do užšieho priestoru a body krivky sa priblížia k sebe. V takejto situácii krivku nechceme rozdeliť, pričom aj v tomto prípade normály smerujú opačným smerom a aj indexová vzdialenosť je väčšia, ako 3. Rozdielom však je, že v situácii v prvom riadku ukazuje normála N_{s_1} smerom k bodu s_2 a normála N_{s_2} smerom k bodu s_1 , ale v druhom prípade už táto podmienka neplatí. Túto odlišnosť budeme využívať aj v algoritme.

Algoritmus spočíva z niekoľkých krokov:

1. Máme maticu P , ktorá bude predstavovať množinu všetkých pixelov. Pixely v ktorých ležia body krivky si označíme najprv hodnotou 0.
2. Prechádzame postupne každý bod krivky a zapisujeme indexy týchto bodov do matice P , pričom za každým skontrolujeme či je hodnota, ktorú chceme prepísať rovná 0, alebo či je väčšia.
 - (a) Ak je hodnota rovná 0, znamená to, že tento pixel ideme označiť indexom bodu prvýkrát, a teda ešte sme nenarazili na iný bod krivky, ktorý by v danom pixeli ležal. V takomto prípade hodnotu pixela v matici P prepíšeme na index bodu, ktorý kontrolujeme.
 - (b) Ak je hodnota väčšia ako 0, tak to znamená, že v danom pixeli leží aj iný bod. V takomto prípade prechádzame na krok 3.
3. Skontrolujeme, rozdiel indexov bodov, ktoré sú adeptami na rozdelenie krivky, aby sme zistili či sú susedné alebo nie. Pokiaľ je rozdiel indexov bodov väčší ako 3, prechádzame na bod 4.
4. Označme si indexy bodov, ktoré sme vybrali, s_1 a s_2 . Budeme počítat vzdialenosti bodov s indexami $\{s_1 - 2, s_1 - 1, s_1, s_1 + 1, s_1 + 2\}$ od bodov $\{s_2 - 2, s_2 - 1, s_2, s_2 + 1, s_2 + 2\}$ a vyberieme body s najmenšou vzdialenosťou. Tým sme prehľadali okolité body, aby sme zistili či medzi nimi nie je dvojica, ktorej vzdialenosť je nižšia ako vzdialenosť bodov γ_{s_1} a γ_{s_2} .
5. Posledným krokom je overenie smerovania normál v bodoch, ktoré boli vybraté v kroku 4. Krivku chceme rozdeliť na dve iba v prípade, že normálové vektory smerujú k sebe, preto preveríme nasledujúcu podmienku

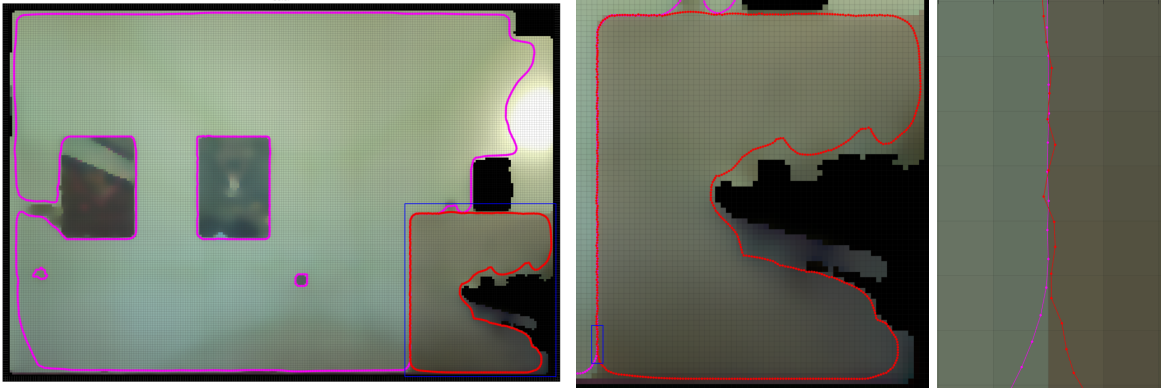
$$|\gamma_{s_1} - \gamma_{s_2}| > |\gamma_{s_1} - (\gamma_{s_2} + \varepsilon N_{s_2})|, \quad (3.69)$$

kde ε je malá hodnota, ktorou prenásobíme normálu N_{s_2} . Pokiaľ je táto podmienka splnená, krivku môžeme rozdeliť na dve.

3.6.3 Spájanie kriviek

Pokiaľ sa k sebe priblížia dve krivky, tak preverujeme, či spĺňajú určité podmienky. Krivky musia mať podobné priemerné hodnoty vlastností v pixeloch, cez ktoré prešli vo všetkých predchádzajúcich časových krokoch. Ďalej preverujeme, či sa krivky nedotýkajú v oblasti hrany nájdenej hranovým detektorom. Obe tieto podmienky prispievajú k tomu, aby sa nespájali krivky, ktoré ležia v iných typoch oblastí. Pre lepšie pochopenie si ukážeme vizualizáciu na obrázku 3.13.

Na ľavom obrázku je ukázaný vývoj kriviek, pričom modrým obdĺžnikom je označený detailnejší výsek, ktorý predstavuje stredný obrázok. Na tomto obrázku môžeme pozorovať, že dve krivky sa dotýkajú a v dôsledku toho, že segmentujú oblasti s rozličnými hodnotami vlastností mračna (v tomto prípade farby) sa nespoja. Na obrázku 3.13 v pravo, ktorý je



Obr. 3.13: Ukážka dvoch kriviek, ktoré sa nespoja v dôsledku nesplnenia podmienok na spojenie.

detailom zo stredného obrázku vyznačeným modrým obdĺžnikom vidíme, že krivky sa naozaj veľmi tesne dotýkajú.

Algoritmus na spájanie sa skladá z niekoľkých krokov:

1. Podobne, ako pri rozdeľovaní krivky, máme maticu P , ktorá bude predstavovať množinu všetkých pixelov, a hodnota bude všade 0. Budeme mať dva cykly pričom vo vonkajšom budeme iterovať cez indexy kriviek a vo vnútornom cez indexy bodov kriviek. V každej iterácii budeme značiť indexy kriviek do matice P podľa toho, v ktorom pixeli ležia body krivky. Pred tým, ako zaznačíme index krivky do matice P , pozrieme sa na hodnotu pixela, ktorý chceme prepísať. V takomto prípade rozlišujeme dve situácie:
 - (a) Ak je táto hodnota rovná 0, to znamená, že tento pixel sme zatiaľ neprepisovali, a teda sme nenarazili na iný bod, ktorý by ležal v tomto pixeli. Podobná situácia nastáva v prípade, že hodnota matice P pre tento pixel je rovná indexu krivky, ktorú preverujeme. To znamená, že tam leží bod rovnakej krivky, a teda nie je potrebné krivku spájať s inou.
 - (b) V prípade, že je hodnota rôzna od 0 a indexu krivky, ktorej prislúcha daný bod, tak to znamená, že v tomto pixeli leží aj bod inej krivky. V takomto prípade prejdeme na krok 2.
2. Pokiaľ v jednom pixeli ležia body dvoch rôznych kriviek, musíme overiť, či krivky nesegmentujú oblasti, v ktorých vlastnosti mračna dosahujú príliš odlišné hodnoty. Indexy kriviek, ktorých body ležia v rovnakom pixeli p_i , budeme označovať c_1 a c_2 . Naším cieľom je zistiť odlišnosť pixelov, ktoré boli navštívené krivkou c_1 od pixelov navštívených krivkou c_2 vo všetkých predchádzajúcich časových krokoch a na základe toho určiť, či sa krivky majú spojiť, alebo nie. Preto budeme pre každú vlastnosť počítať nasledujúcu vzdialenosť

$$d_l = \left| \overline{I_{\Omega^n, c_1}^l} - \overline{I_{\Omega^n, c_2}^l} \right| \cdot (1 - g_1(p_i)). \quad (3.70)$$

Hodnoty $\overline{I_{\Omega^n, c_1}^l}$ a $\overline{I_{\Omega^n, c_2}^l}$ sú priemerné hodnoty vlastnosti l v pixeloch, ktoré boli vo všetkých predchádzajúcich časových krokoch navštívené krivkami c_1 a c_2 a počítame ich vzťahom (3.60). Absolútnu hodnotu z ich rozdielu ďalej násobíme hodnotou $(1 - g_1(p_i))$. Táto hodnota znižuje výslednú hodnotu d_l v prípade, že pixel p neleží na hranici. Ak je hodnota d_l vysoká tak to znamená, že krivky prešli cez oblasti s inými hodnotami vlastnosti l , a preto, takéto krivky nechceme spájať. Hodnotu d_l budeme porovnávať s parametrom μ_l , ktorý predstavuje prahovú hodnotu vypočítanú podľa vzťahov zo sekcie 2.3.3. Keďže máme k dispozícii viac vlastností, aj v tomto prípade ich budeme spájať.

Aby boli krivky spojené musí platiť táto nerovnica

$$\sum_l d_l \leq \sum_l 0.5 \cdot \mu_l. \quad (3.71)$$

Pokiaľ je splnená nerovnica (3.71), tak prechádzame na krok 3.

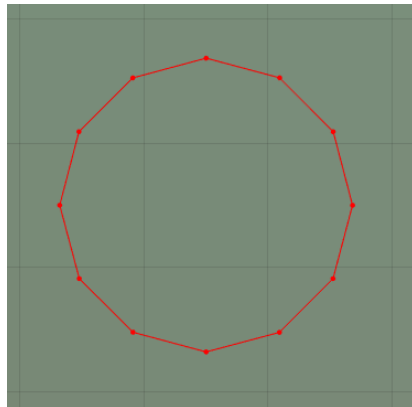
3. Podobným spôsobom, ako v kroku 4 algoritmu na rozdeľovanie kriviek, budeme prehľadávať susedné body a budeme hľadať tie, ktoré sú k sebe najbližšie.
4. Vzďialenosť bodov, ktoré sme vybrali v kroku 3 budeme označovať h_s . Ďalej musí platiť nasledujúca podmienka

$$h_s \leq 0.5 \cdot (\overline{h_{c_1}} + \overline{h_{c_2}}). \quad (3.72)$$

Hodnoty $\overline{h_{c_1}}$ a $\overline{h_{c_2}}$ sú priemerné hodnoty vzdialeností susedných bodov na krivkách c_1 a c_2 . Touto podmienkou overujeme, či je vzdialenosť bodov, v ktorých chceme krivky spojiť, menšia, ako priemerná vzdialenosť bodov na oboch krivkách. Pokiaľ je aj táto podmienka splnená môžeme krivky spojiť.

3.7 Vkládanie počiatočných kriviek

Počiatočné krivky vkladáme v tvare kružnice, pričom polomer je rovný dĺžke hrany pixela. Ukážku počiatočnej krivky môžeme vidieť na obrázku 3.14.

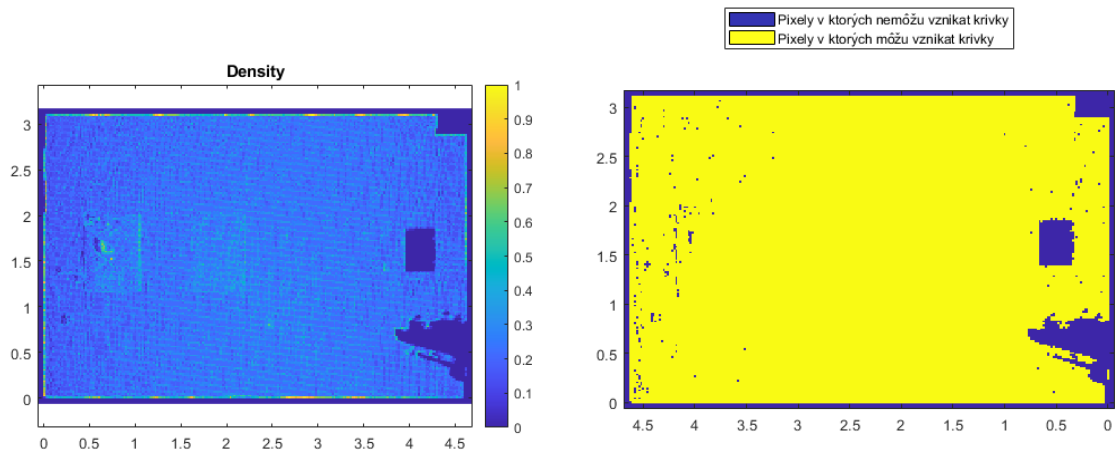


Obr. 3.14: Vizualizácia počiatočnej krivky.

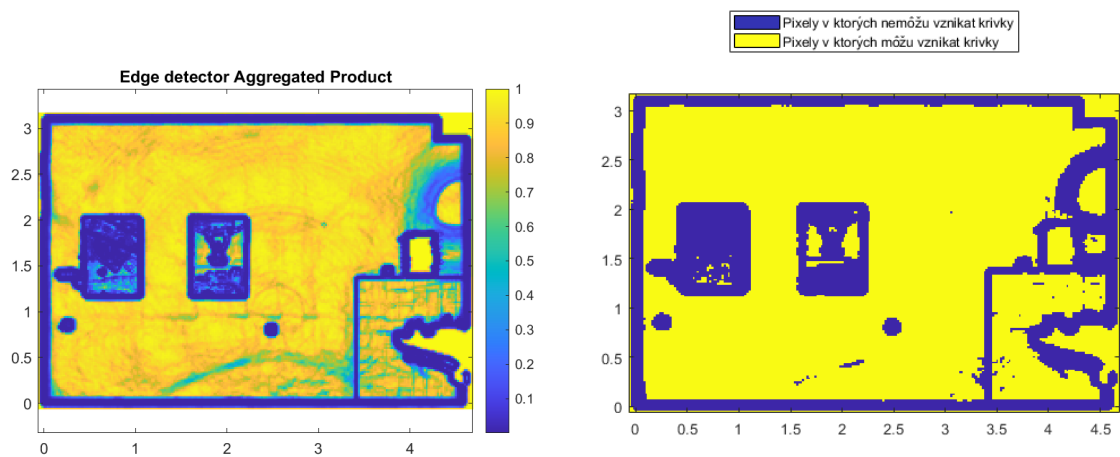
Stred kružnice leží v strede vybraného pixela a tento pixel vyberáme náhodne, z množiny pixelov, ktoré spĺňajú nasledujúce podmienky.

1. Hustota bodov ležiacich v pixeli musí byť väčšia, ako nami určená hodnota. Keďže regresná rovina obsahuje aj pixely v ktorých neleží ani jeden bod, tak sa chceme vyhnúť situácii, kedy by sa počiatočná krivka začala vyvíjať mimo oblasti, ktorú chceme segmentovať. Pre lepšiu predstavu si ukážeme vizualizáciu hustoty, vedľa vizualizácie pixelov spĺňajúcich podmienku hustoty na obrázku 3.15.
2. Krivka nesmie ležať na hrane alebo v jej blízkosti, pretože by to mohlo zastaviť, alebo spomaliť vývoj krivky. Preto vyberáme také pixely, v ktorých je hodnota agregovaného hranového detektora väčšia ako 0.7. Na obrázku 3.16, môžeme vidieť obrázok znázorňujúci agregovaný hranový detektor a vedľa neho vizualizáciu pixelov spĺňajúcich podmienku hranového detektora.

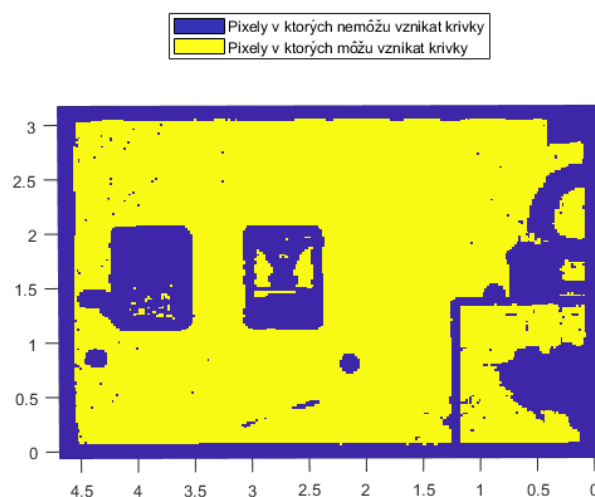
Spojením tých dvoch podmienok získavame výslednú množinu pixelov, z ktorých vyberáme počiatočné krivky. Vizualizáciu môžeme vidieť na obrázku 3.17.



Obr. 3.15: Pixely spĺňajúce kritériá na vloženie počiatočnej krivky podľa podmienky hustoty.



Obr. 3.16: Pixely spĺňajúce kritériá na vloženie počiatočnej krivky podľa podmienky hranového detektora.



Obr. 3.17: Pixely spĺňajúce kritériá na vloženie počiatočnej krivky.

V prípade, že na začiatku vývoja vkladáme viac ako jednu počiatočnú krivku, tak z množiny všetkých možných počiatočných pixelov odstránime pixely, v ktorých leží vložená počia-

točná krivka a aj okolité pixely.

3.8 Zastavovacie kritérium

V každom časovom kroku vývoja krivky kontrolujeme, ako veľmi sa krivka pohybuje. Pokiaľ krivka vykonáva iba veľmi malý pohyb, tak chceme ukončiť jej evolúciu. Na to používame nasledujúce zastavovacie kritérium

$$\max_{i=1,\dots,m} (\|\gamma_i^n - \gamma_i^{n-2}\|) \leq \varepsilon. \quad (3.73)$$

V každom časovom kroku počítame dĺžku krokov, ktorú vykonali jednotlivé body. Z nich vyberieme maximálnu hodnotu a pokiaľ je táto hodnota menšia, ako nami určený parameter ε , tak ukončíme vývoj krivky.

3.9 Postup celého algoritmu

V tejto podkapitole si vysvetlíme postupnosť krokov algoritmu, využívajúc metódy uvedené v kapitole 3.

1. Vypočítame pixelovú reprezentáciu vlastností mračna podľa postupu v podkapitole 3.2.
2. Na základe pixelových reprezentácií vlastností vypočítame hranové detektory podľa postupu z podkapitoly 3.2.2 a následne agregovaný hranový detektor podľa podkapitoly 3.5.2.
3. Z histogramov pixelových reprezentácií vlastností si vypočítame hraničné hodnoty μ podľa postupu z podkapitoly 2.3.3.
4. Vytvoríme množinu bodov, v ktorých môžu vzniknúť krivky a vygenerujeme N počiatočných kriviek podľa postupu v podkapitole 3.7
5. V časovom for cykle prebieha nasledovný postup krokov:
 - (a) Spustí sa algoritmus topologických zmien z podkapitoly 3.6, pričom časti algoritmu prebiehajú v nasledujúcom poradí:
 - i. Rozdelenie kriviek 3.6.2.
 - ii. Spájanie kriviek 3.6.3.
 - iii. Pridávanie odoberanie bodov 3.6.1.
 - (b) Vo for cykle cez všetky vyvíjajúce sa krivky prebiehajú nasledujúce kroky:
 - i. Vypočítajú sa hodnoty funkcie homogenity pre všetky vlastnosti podľa postupu zo sekcie 3.4.2 a agregovaná funkcia homogenity podľa postupu zo sekcie 3.5.3. Ďalej vypočítame hodnoty normálovej rýchlosti riadenej vonkajším vplyvom β_E podľa postupu zo sekcie 3.4.2.
 - ii. Na základe vypočítaných hodnôt β_E vypočítame krivostnú regularizáciu podľa sekcie 3.4.1.
 - iii. Vypočítame hodnoty tangenciálnej rýchlosti podľa podkapitoly 3.3.
 - iv. Uložíme si maximálnu hodnotu β do premennej $\beta_{max} = \max_{i=1,\dots,m} \beta_i$.
 - (c) Vypočítame veľkosť časového kroku Δt podľa vzťahu

$$\Delta t = \frac{h_d}{\max_{j=1,\dots,N} \beta_{max,j}}, \quad (3.74)$$

kde $h_d = \frac{h}{2}$ a $\max_{j=1,\dots,N} \beta_{max,j}$ je maximálna hodnota normálovej rýchlosti zo všetkých bodov kriviek.

- (d) Opäť vo for cykle cez všetky vyvíjajúce sa krivky prebiehajú nasledujúce kroky:
- i. Zostavíme a vyriešime systém rovníc podľa postupu z podkapitoly 3.1.
 - ii. Overíme či evolúciu krivky netreba zastaviť podľa zastavovacieho kritéria z podkapitoly 3.8.
- (e) Overíme či sa ešte nejaká krivka vyvíja, ak nie, zistíme koľko percent pixelov, na ktorých môžu vzniknúť počiatočné krivky podľa kritérií z podkapitoly 3.7, nebolo navštívených žiadnou z kriviek. Ak je toto percento väčšie, ako nami zadaná hodnota, tak podobným spôsobom, ako na začiatku, vygenerujeme množinu pixelov, do ktorých budeme ďalej vkladať krivky s tým rozdielom, že neuvažujeme pixely navštívené inými krivkami vo všetkých predchádzajúcich časových krokoch a novovzniknuté krivky pokračujú v postupnosti algoritmu od kroku 5(a). Pokiaľ je toto percento dostatočne veľké, tak ukončujeme algoritmus na vývoj kriviek.

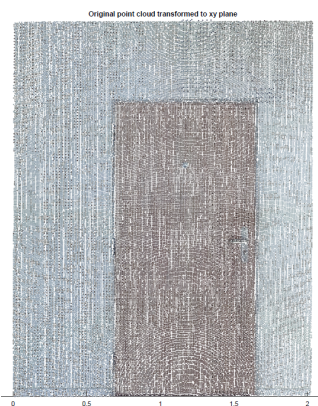
Kapitola 4

Numerické experimenty

V tejto kapitole si ukážeme niekoľko regresných rovín, ktoré sú výstupom prvého kroku segmentácie. Na týchto rovinách budeme vidieť aplikácie metód, ktoré sú popísané v kapitole 3. V prvom numerickom experimente, na ktorom budeme demonštrovať funkčnosť algoritmu, použijeme na začiatok iba jednu počiatočnú krivku a opíšeme aplikáciu metód na vkladanie počiatočných kriviek, spájanie hranových detektorov a výpočet prahových hodnôt funkcií homogenity. V ďalších numerických experimentoch budeme používať niekoľko počiatočných kriviek, pričom uvidíme, ako funguje algoritmus topologických zmien, vďaka ktorému sa budú krivky spájať a rozdeľovať.

4.1 Stena s dverami

Ako prvý numerický experiment použijeme jednoduchú stenu, v ktorej ležia dvere. Túto regresnú rovinu chceme rozdeliť na dva segmenty, kde jeden segment budú tvoriť body prislúchajúce stene a druhý segment body prislúchajúce dverám. Mračno bodov zafarbené použitím farebných kanálov R, G a B môžeme vidieť na obrázku 4.1.

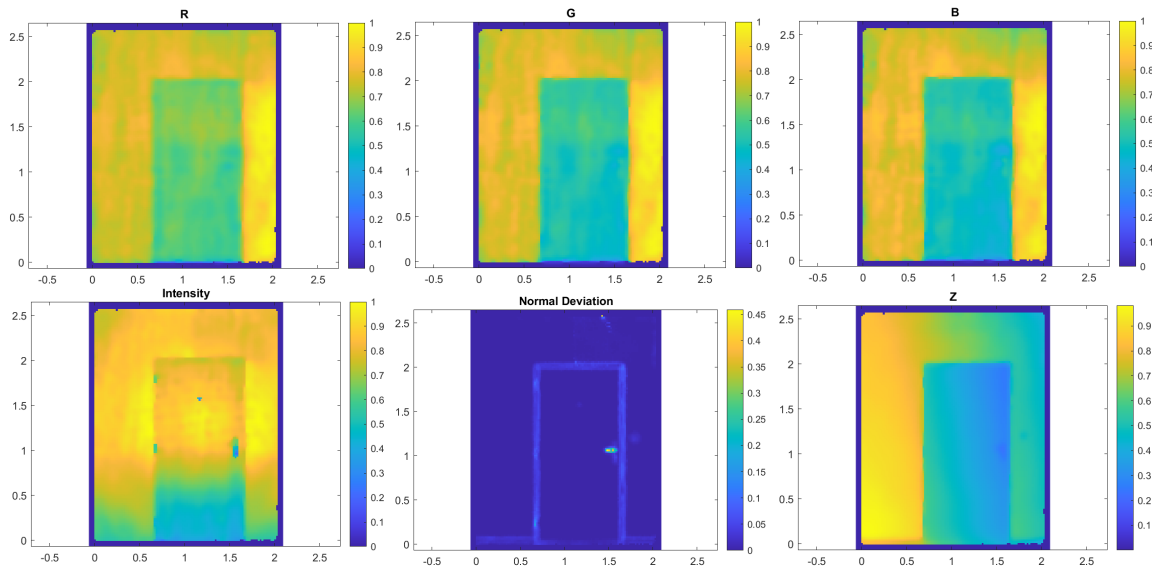


Obr. 4.1: Mračno bodov vyfarbené farebnými kanálmi R, G a B.

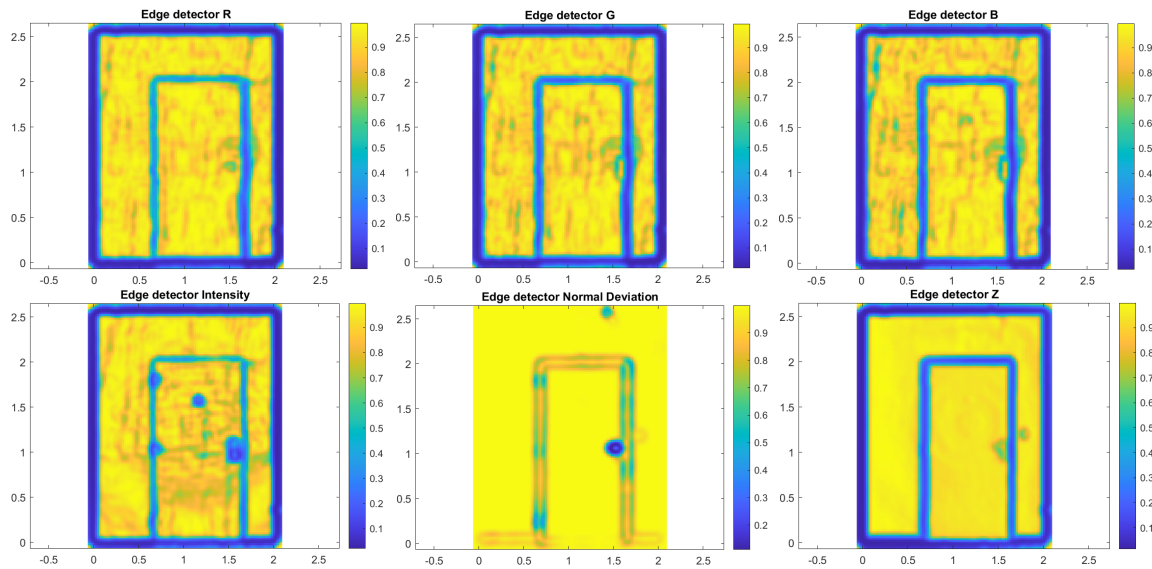
Prvým krokom algoritmu je výpočet pixelových reprezentácií jednotlivých vlastností spôsobom popísaním v sekcii 3.2. K dispozícii máme celkovo 6 vlastností, ktorými sú: farebné kanály R, G a B, intenzita, lokálna odchýlka normálových vektorov a vzdialenosť bodov mračna od regresnej roviny, ktorú vo vizualizáciách nazývame Z . Pixelovú reprezentáciu všetkých týchto vlastností môžeme vidieť na obrázku 4.2.

Ďalej vypočítame hranové detektory pre pixelové reprezentácie všetkých vlastností mračna, ktoré môžeme vidieť na obrázku 4.3.

Po výpočte hranových detektorov počítame agregované hranové detektory podľa postupu opísaného v sekcii 3.5.2. V našom algoritme sme implementovali oba uvedené prístupy, ale v

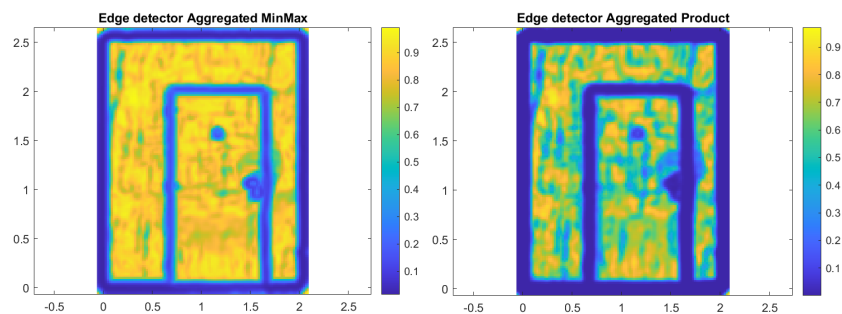


Obr. 4.2: Pixelová reprezentácia vlastností mračna bodov.



Obr. 4.3: Hranové detektory všetkých vlastností.

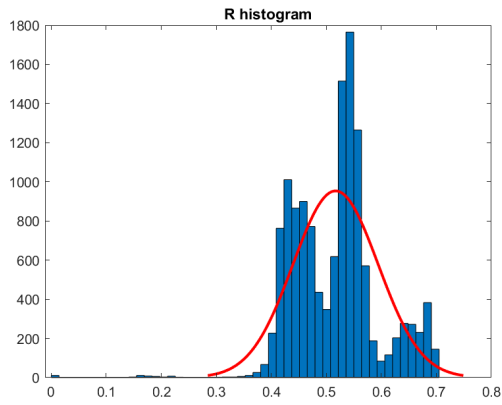
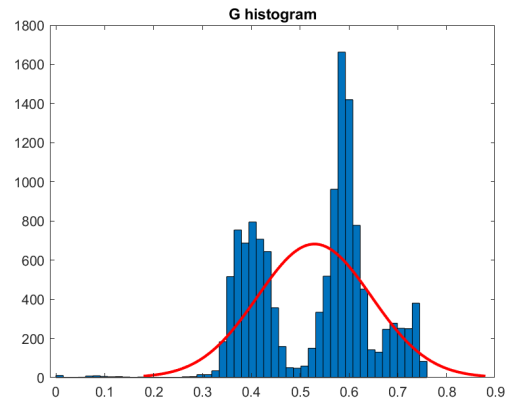
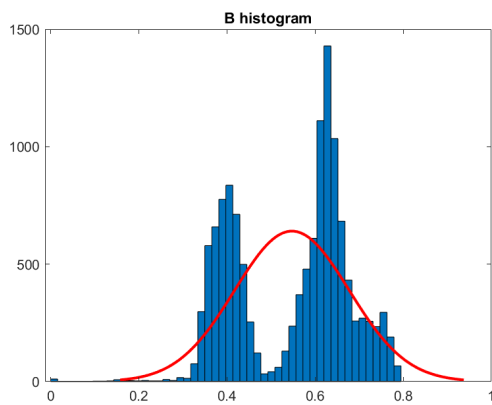
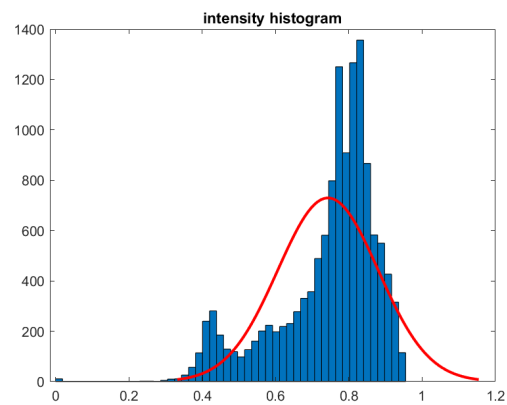
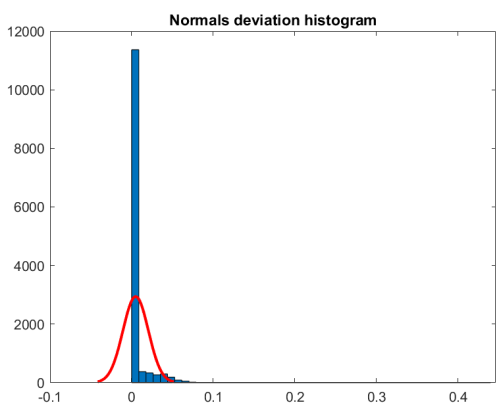
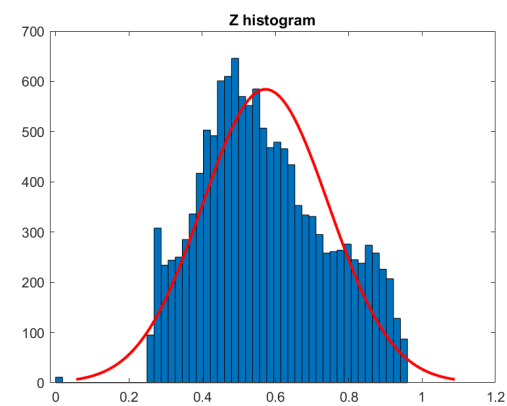
konečnom dôsledku používame iba jeden z nich, v tomto prípade je to súčinnový agregovaný hranový detektor. Oba tieto hranové detektory sú vizualizované na obrázku 4.4.



Obr. 4.4: Agregované hranové detektory.

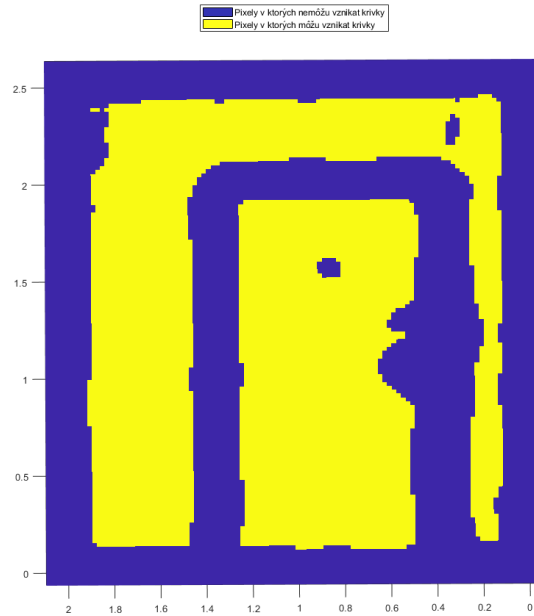
Ďalším krokom postupu je výpočet prahových hodnôt μ , ktoré využívame počas vývoja

vo výpočte funkcii homogenity a overení spájania kriviek. Pre každý histogram vypočítame funkciu normálneho rozdelenia, a na základe jej štandardnej odchýlky σ počítame prahové hodnoty μ . Obrázky histogramov spolu s funkciou normálneho rozdelenia a prahovou hodnotou μ môžeme vidieť vo vizualizácii 4.5.

(a) $\mu = 0.2174$ (b) $\mu = 0.2566$ (c) $\mu = 0.27644$ (d) $\mu = 0.3313$ (e) $\mu = 0.0121$ (f) $\mu = 0.3341$

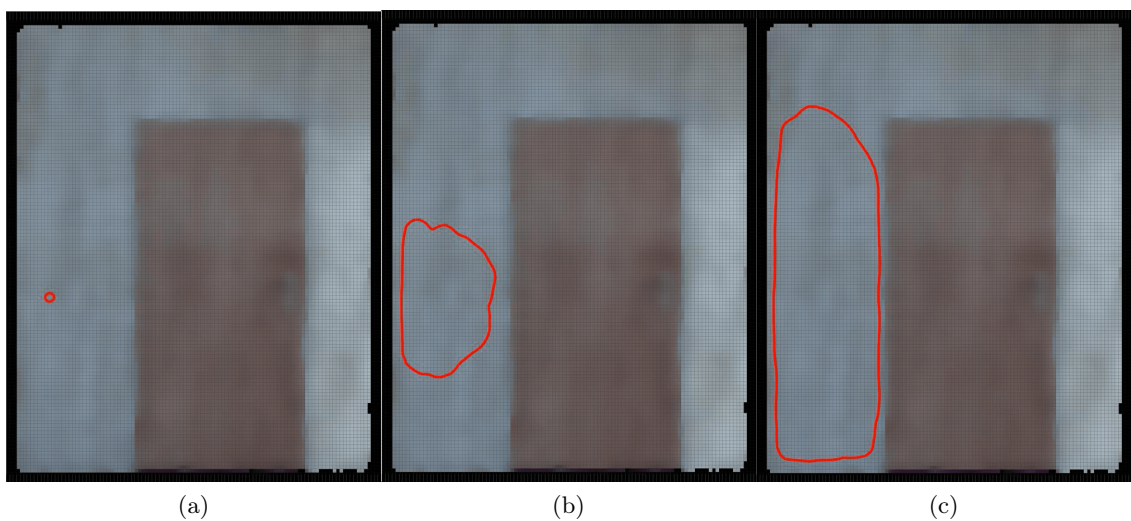
Obr. 4.5: Histogramy vlastností.

Posledným krokom pred tým, ako začneme vyvíjať krivky, je určiť pixely, do ktorých môžeme vložiť počiatočné krivky. Použitím hranového detektora a hustoty mračna dostaneme pomocou postupu z podkapitoly 3.7 množinu pixelov, do ktorých môžu byť vložené počiatočné krivky. Vizualizáciu tejto množiny je možné vidieť na obrázku 4.6.

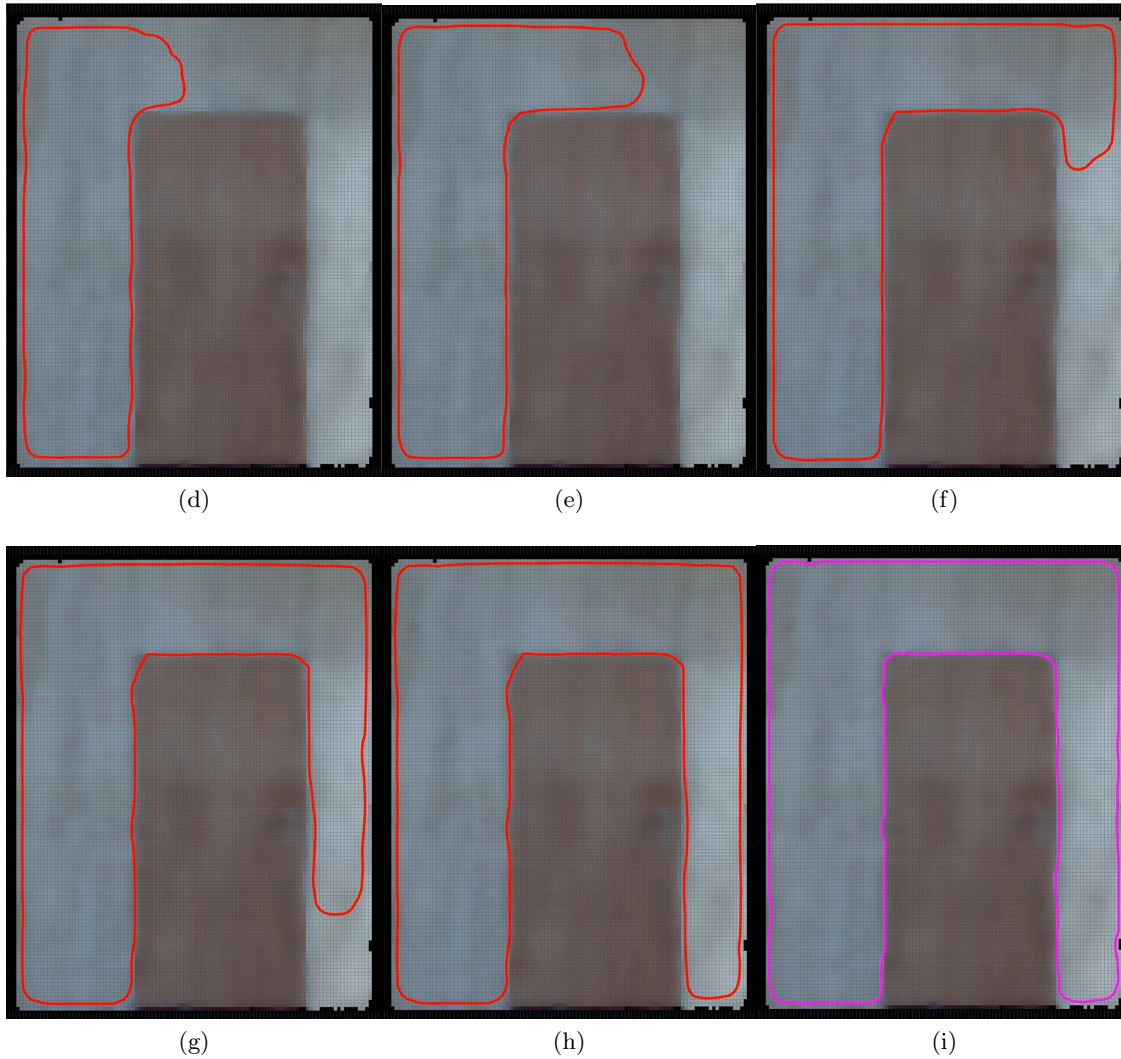


Obr. 4.6: Vizualizácia pixelov vhodných na vloženie počiatočných kriviek.

Teraz už máme všetko pripravené, aby sme mohli vložiť prvú krivku a začali ju vyvíjať. Počas jedného časového kroku algoritmus vypočíta hodnoty tangenciálnej rýchlosti α podľa postupu z podkapitoly 3.3, normálovej rýchlosti podľa podkapitoly 3.4 a následne zostaví a rieši systém rovníc podľa semi-implicitnej schémy uvedenej v podkapitole 3.1. Vyriešením systému získavame polohu bodov v nasledujúcom časovom kroku. Niekoľko časových krokov evolúcie krivky vidíme vykreslených na obrázku 4.7.

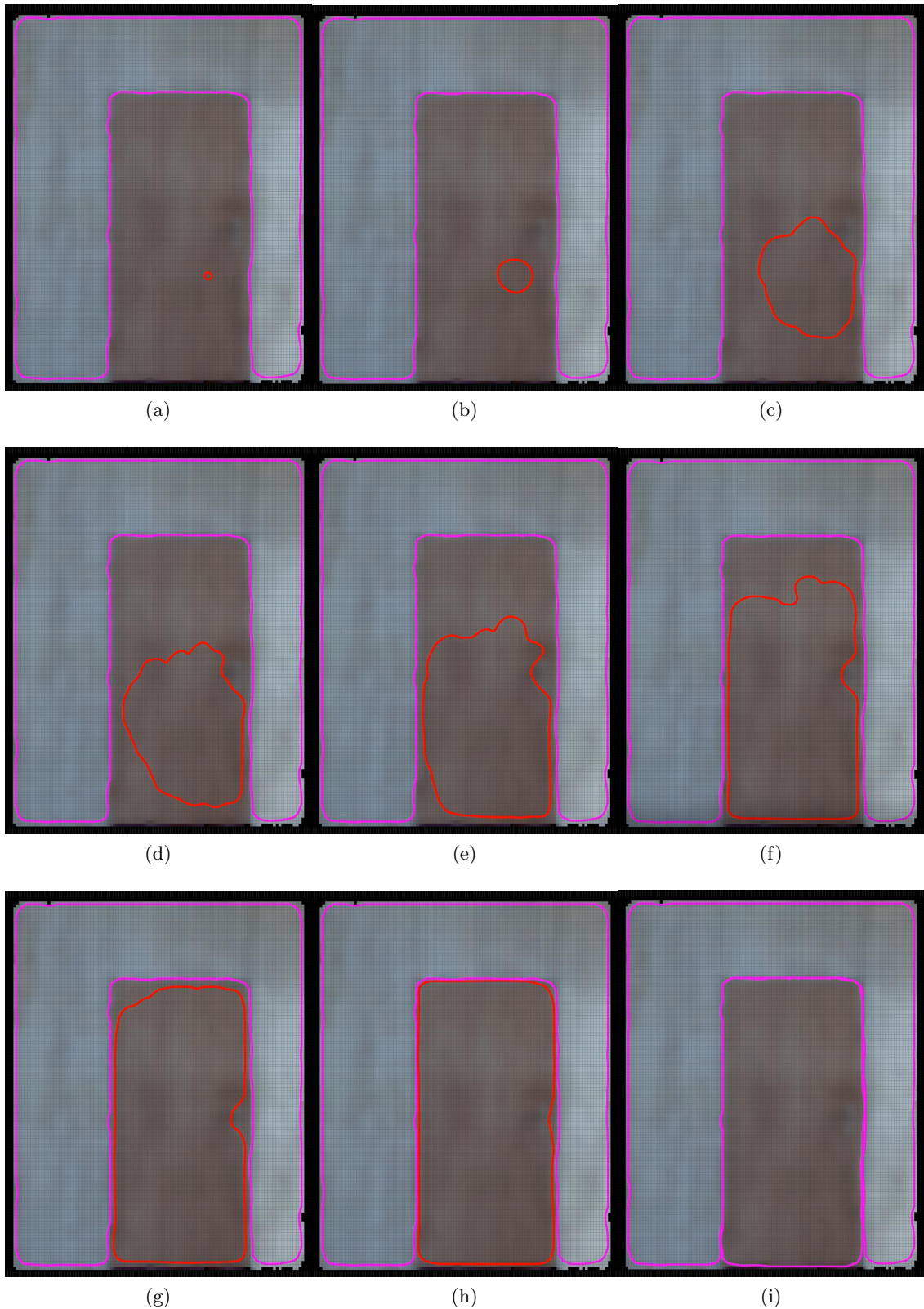


Obr. 4.7: Evolúcia krivky.



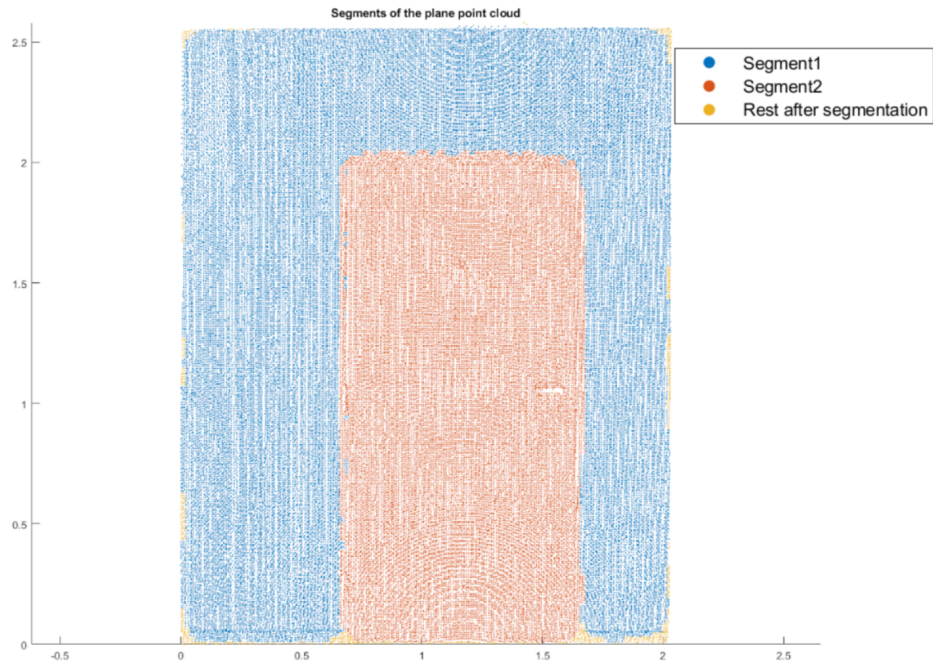
Obr. 4.7: Evolúcia krivky.

Na obrázku 4.8:(i) môžeme krivku vidieť sfarbenú na fialovo, pričom na ostatných obrázkoch je sfarbená na červeno. To značí, že parameter λ , ktorý vystupuje v rovniciach na výpočet tangenciálnej a normálovej rýchlosti bol prepnutý z hodnoty 0.5 na hodnotu 1 v dôsledku spomalenia rýchlosti krivky. Po prepnutí parametra λ na hodnotu 1 krivku vyvíja už iba gradient hranového detektora podľa rovnice (3.59), a po maximálne 30 časových iteráciách zastavujeme vývoj krivky. Na tomto obrázku je vidieť, že zostala ešte značná časť nevysegmentovaná, preto algoritmus vloží novú krivku do pixelov vhodných na vkladanie nových kriviek, ktoré neboli navštívené v predchádzajúcich časových krokoch žiadnou krivkou. Vloženie ďalšej krivky v oblasti dverí a jej evolúciu môžeme vidieť na obrázkoch 4.8.



Obr. 4.8: Vloženie ďalšej počiatočnej krivky a jej vývoj.

Po ukončení evolúcie druhej krivky vidíme, že už je len veľmi malé percento pixelov, ktoré krivky nenavštívili. Preto ukončujeme evolúciu kriviek na tejto stene. Vizualizáciu segmentov vidíme na obrázku 4.9. Výstupom krivkovej segmentácie je takéto mračno bodov rozdelené

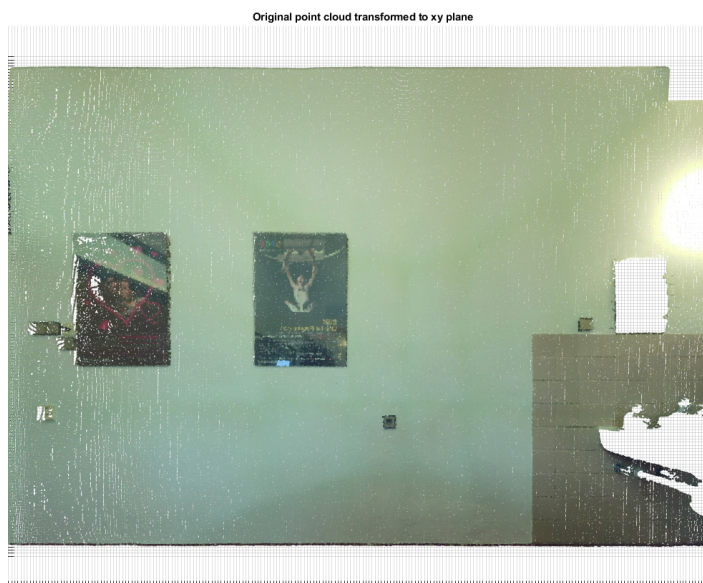


Obr. 4.9: Mračno vysegmentovaných bodov.

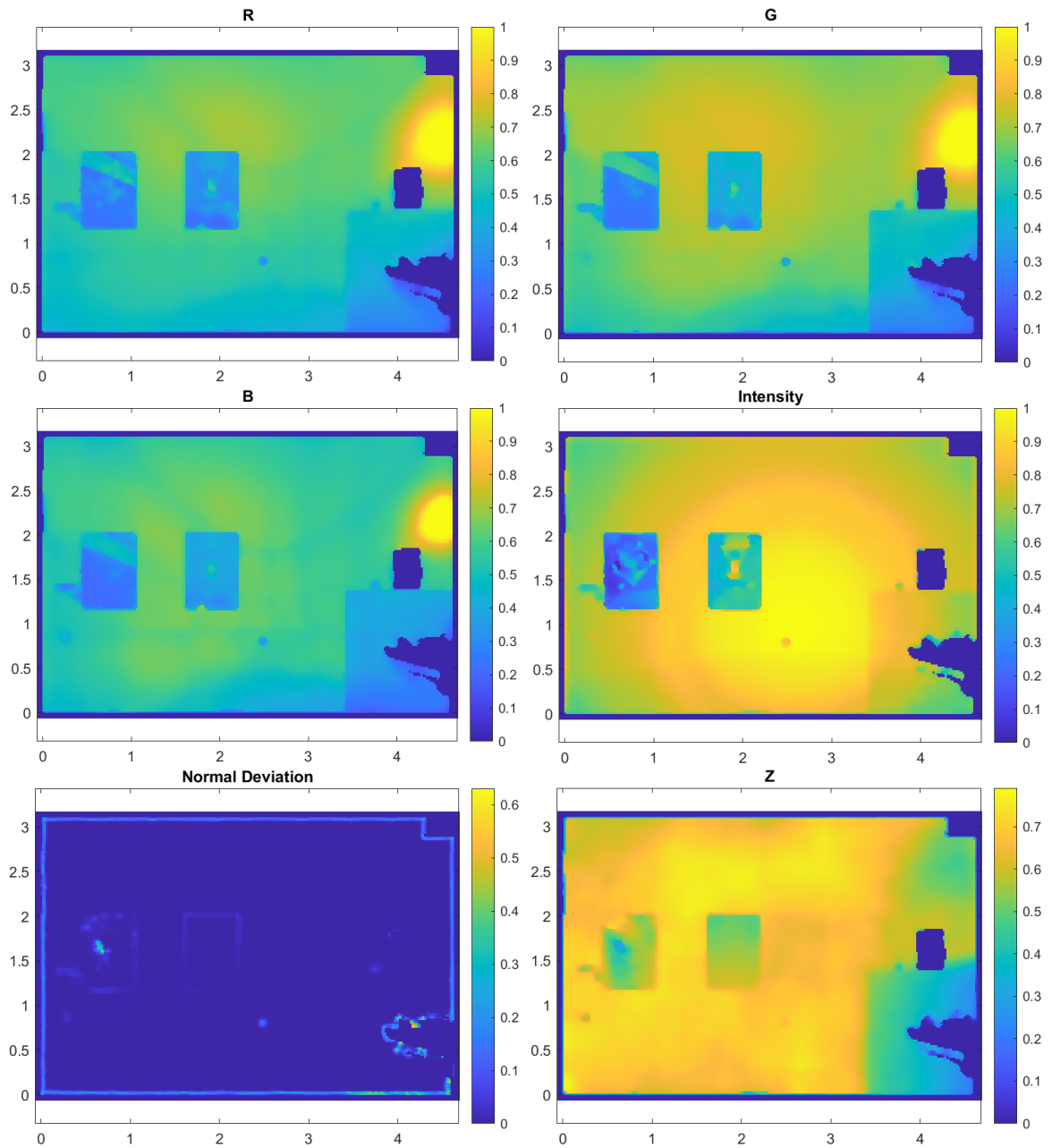
na segmenty. Tieto segmenty sa v rámci celého projektu (ktorého súčasťou táto diplomová práca) budú ďalej porovnávať s objektami v BIM modeli.

4.2 Stena s plagátmi a obkladom

Ďalším numerickým experimentom, na ktorom predvedieme implementovaný algoritmus, bude stena, ktorú sme používali v kapitole 3, na vizualizáciu niektorých častí nášho algoritmu. Na tejto stene sa nachádzajú dva plagáty a v pravom dolnom rohu je obloženie. Tiež sú tu elektrické zásuvky a v pravom hornom rohu je miesto, na ktoré počas skenu miestnosti svietila lampa. Vizualizáciu mračna pomocou R, G a B kanálov farby môžeme pozorovať na obrázku 4.10. Pixelové reprezentácie jednotlivých vlastností sú na obrázkoch 4.11.

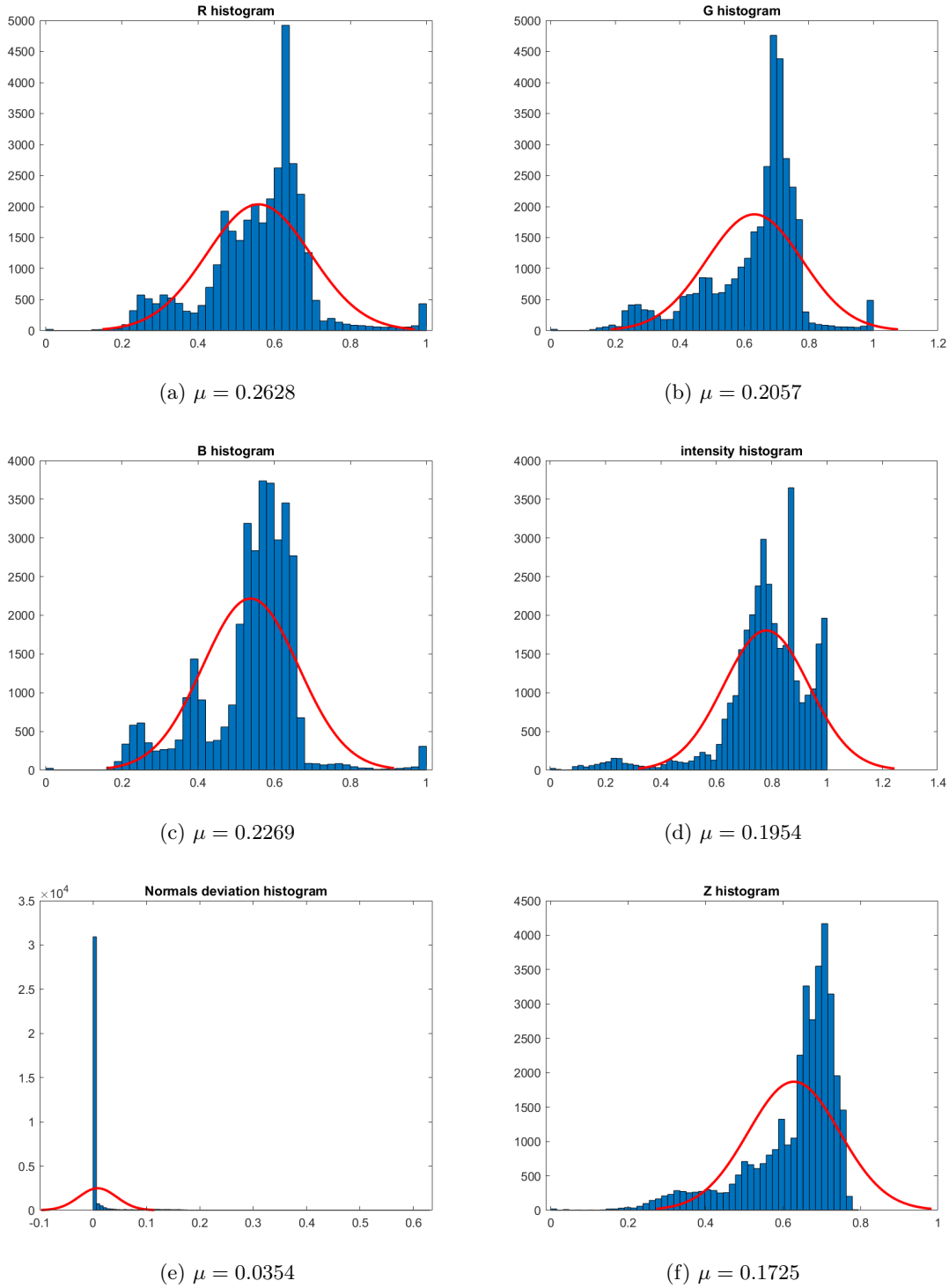


Obr. 4.10: Mračno bodov.



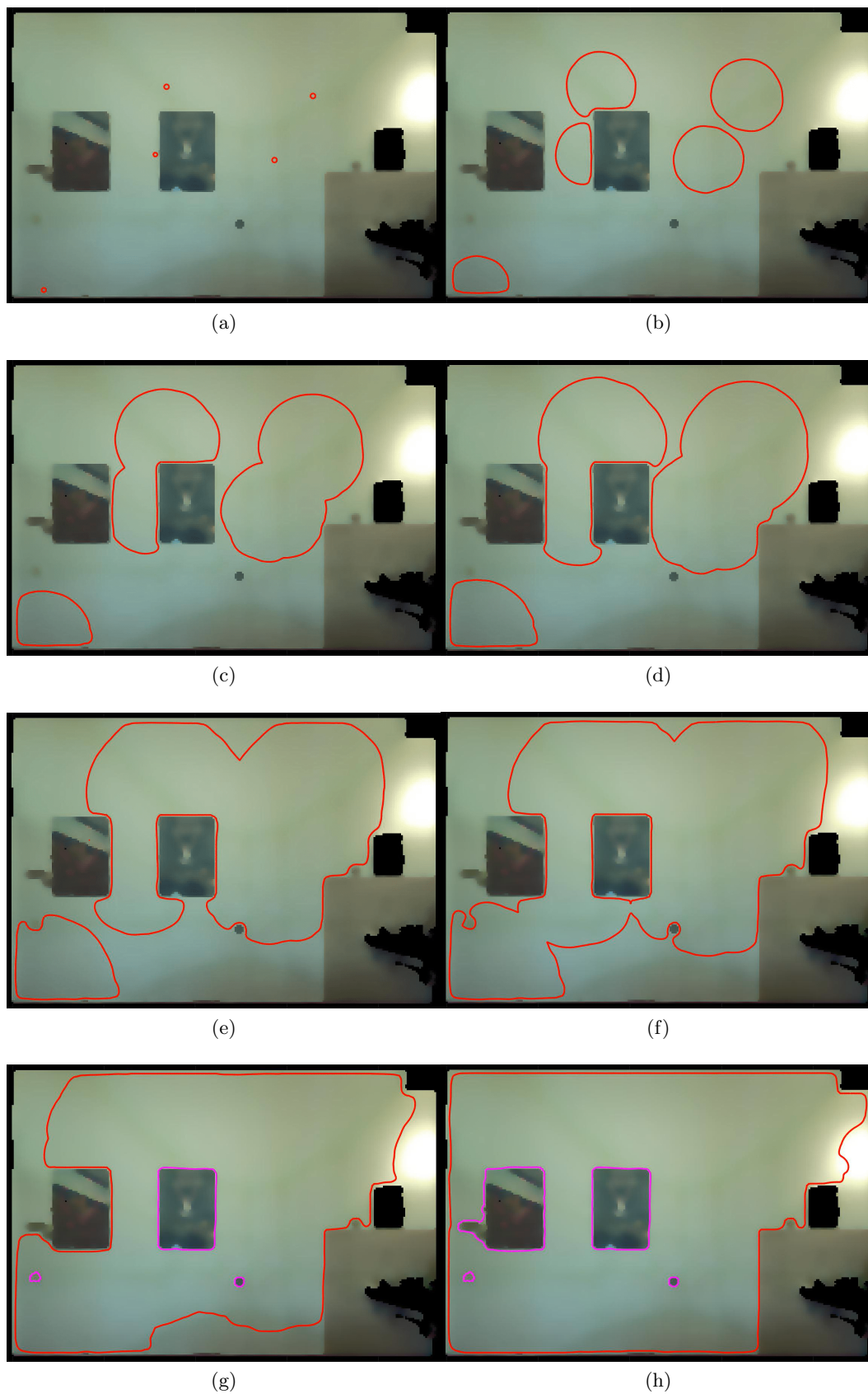
Obr. 4.11: Pixelová reprezentácia vlastností mračna bodov.

Hranové detektory pre všetky vlastnosti sme si ukázali na obrázku 3.10 a agregované hranové detektory na obrázku 3.11 v sekcii 3.5.2. Dalším krokom je vypočítať prahové hodnoty z histogramov vlastností. Vizualizácie histogramov a ich funkcií normálneho rozdelenia môžeme vidieť na vizualizácii 4.12.

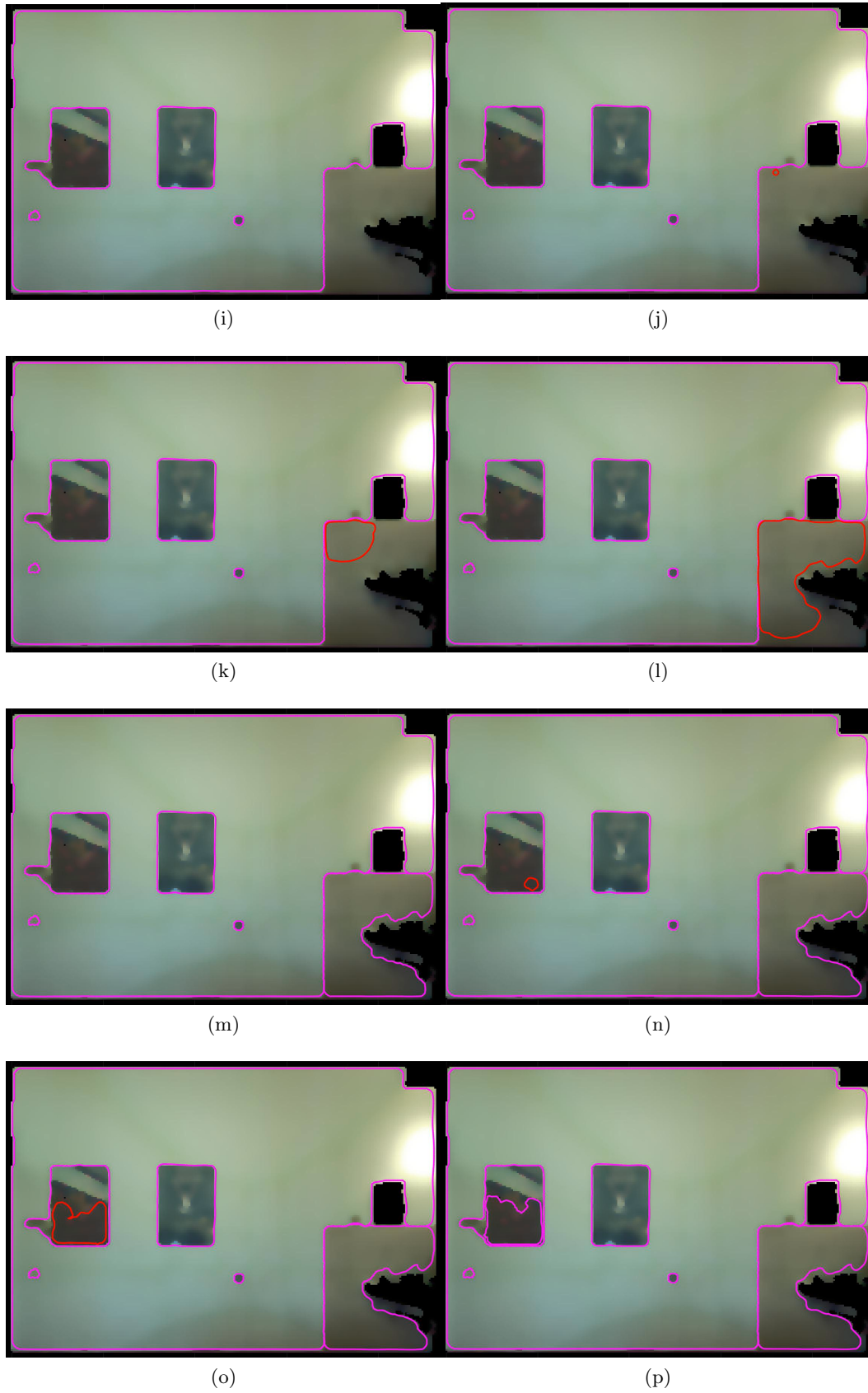


Obr. 4.12: Histogramy vlastností.

Ďalším krokom algoritmu je vypočítať, do ktorých pixelov môžeme vkladať počiatkové krivky. V sekcii 3.7 vysvetľujeme postup výpočtu takýchto pixelov za použitia vizualizácii práve pre túto stenu. Vizualizáciu výslednej množiny pixelov môžeme vidieť na obrázku 3.17. Teraz už máme pripravené všetko na to, aby sme vložili počiatkové krivky a začali ich vyvíjať. V tomto numerickom experimente začneme vyvíjať hneď niekoľko počiatkových kriviek, aby sme sa dostali rýchlejšie k výslednej segmentácii. Vizualizáciu segmentácie môžeme vidieť na obrázkoch 4.13.

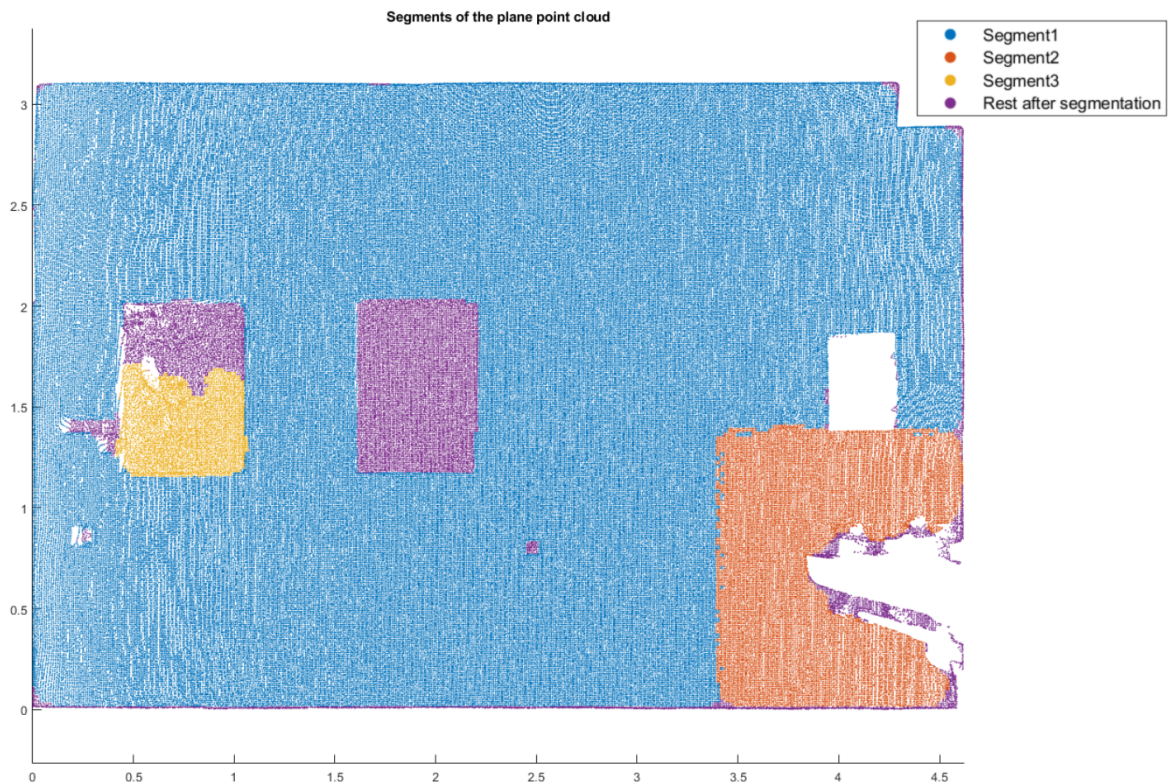


Obr. 4.13: Vývoj kriviek.



Obr. 4.13: Vývoj kriviek.

Môžeme vidieť, že počas vývoja pracoval aj algoritmus na spájanie a rozdeľovanie. Napríklad už medzi obrázkami (b) a (c) je vidieť, že prebehli dve spojenia kriviek a následne z obrázka (d) na (e) je vidieť, že sa aj tieto dve krivky spojili. V obrázku (f) vidíme, že táto krivka obkolesila plagát ležiaci v stene a následne sa musela rozdeliť na dve krivky. Podobný scenár sa dial aj počas prechodu na obrázok (g) kde krivka obkolesila dve zásuvky ležiace v spodnej časti steny a krivka sa rozdelila na viac kriviek. Ďalej vývoj pokračoval, až kým celá oblasť, do ktorej boli vložené počiatkové krivky nebola vysegmentovaná, pričom táto časť segmentácie končí na obrázku (i). Avšak vidíme, že nám ešte ostali homogénne miesta, do ktorých následne vkladáme krivku na obrázku (j). Ako bolo spomínané, táto časť predstavuje oblasť, na ktorej je obklad. Aj na obrázkoch 4.11 je vidieť, že povrch má inú farbu, intenzitu a podľa vizualizácie vlastnosti Z, ktorá značí vzdialenosť od regresnej roviny vidíme, že táto časť je vystúpenejšia, čo značí rozdielny typ oblasti z hľadiska vlastností mračna. Preto nechceme novovzniknutú krivku na tejto oblasti spojiť s krivkou, ktorá túto oblasť obklopila v predchádzajúcich časových krokoch, preto na obrázkoch (k), (l) a (m), na ktorých je vidieť vývoj novovzniknutej krivky vidíme, že krivky sa v dôsledku odlišnosti vlastností nespoja. Podobný prípad nastáva aj počas vývoja ďalšej vlozenej krivky do vnútra jedného z plagátov ležiacich v stene na obrázkoch (n), (o) a (p). Posledným krokom je vytvorenie segmentov, ktoré môžeme vidieť na vizualizácii 4.14.

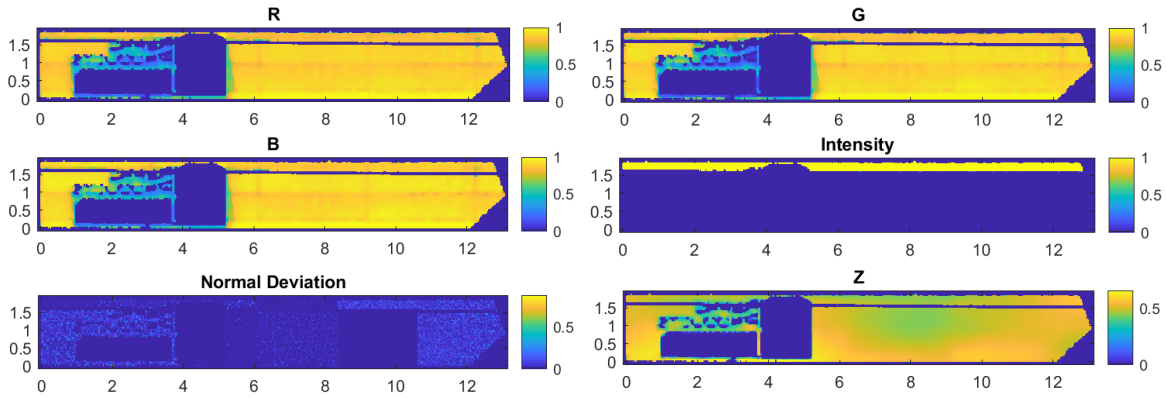


Obr. 4.14: Mračno bodov.

4.3 Predná stena haly

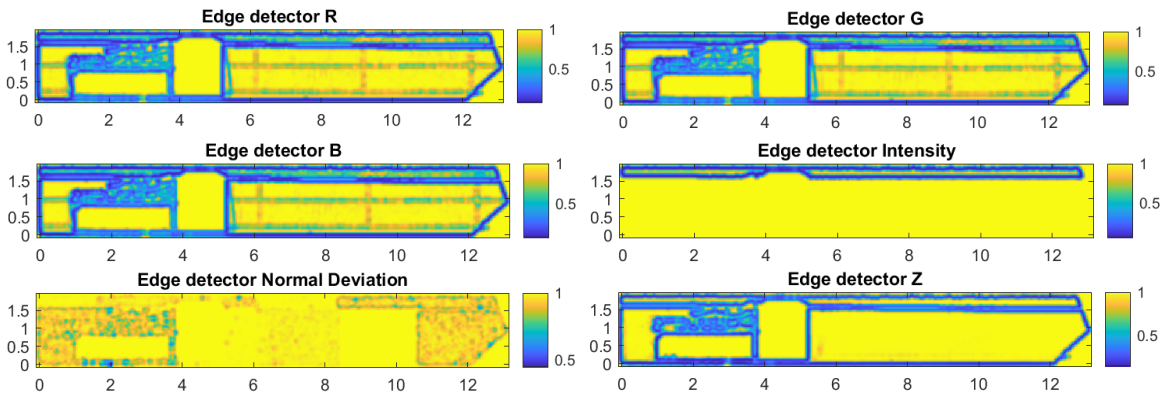
Ďalším numerickým experimentom je jedna z vonkajších stien výrobnéj haly. Celé mračno bodov naskenovanej haly môžeme vidieť na obrázku 1.4 aj s vizualizáciou regresnej roviny. Mračno bodov prislúchajúce regresnej rovine je na obrázku 1.5.

Rovnako, ako v predchádzajúcich numerických experimentoch, je prvým krokom vytvorenie pixelovej reprezentácie vlastností mračna, ktoré vidíme na obrázku 4.15.



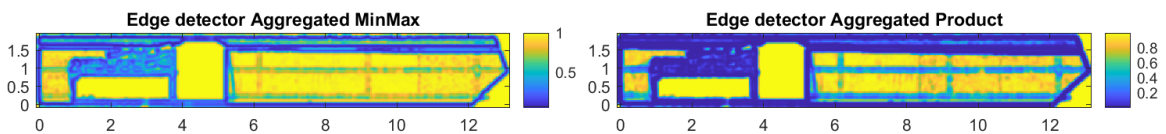
Obr. 4.15: Pixelová reprezentácia vlastností mračna bodov.

Následne počítame hranové detektory a z nich agregované hranové detektory, ktoré môžeme vidieť na obrázkoch 4.16 a 4.17.



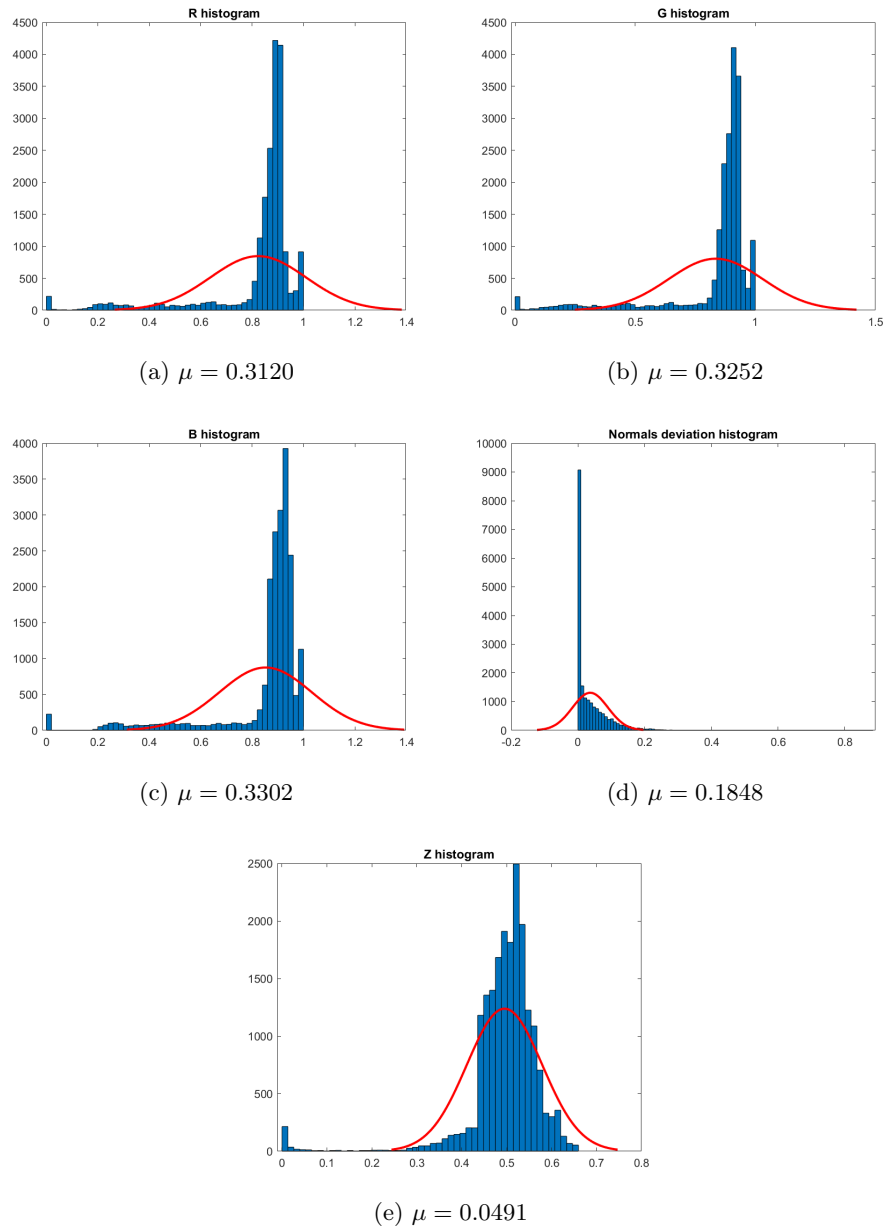
Obr. 4.16: Hranové detektory všetkých vlastností.

Na pixelovej reprezentácii a hranovom detektore pre intenzitu môžeme vidieť, že dáta sú neúplné, preto túto vlastnosť v tomto prípade nebudeme používať pri vývoji kriviek ani na výpočet agregovaného hranového detektora.



Obr. 4.17: Agregované hranové detektory.

Na základe pixelových reprezentácií vlastností vypočítame z histogramov prahové hodnoty.



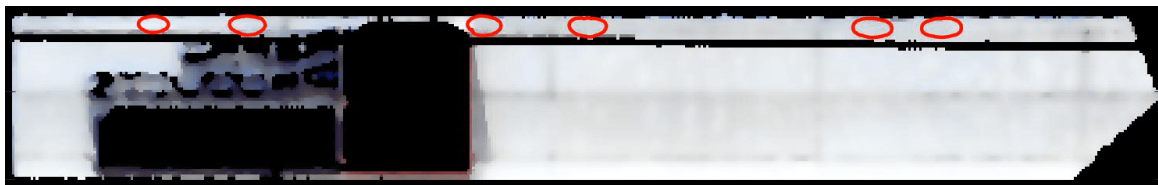
Obr. 4.18: Histogramy vlastností.

Ďalej vypočítame pixely vhodné na vloženie počiatočnej krivky, ktoré sú vizualizované na obrázku 4.19.

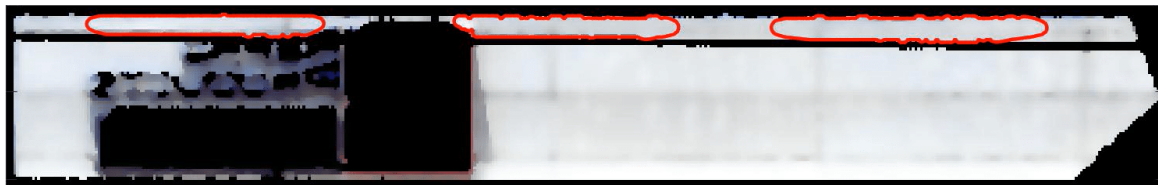


Obr. 4.19: Vizualizácia pixelov vhodných na vloženie počiatočných kriviek.

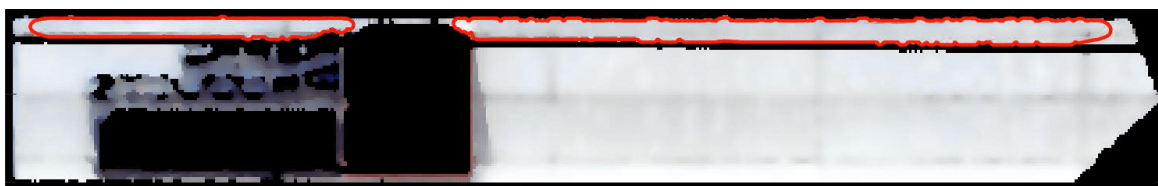
Keď už máme všetko pripravené môžeme vložiť počiatočné krivky a začať s evolúciou, ktorá je vizualizovaná na obrázku 4.20.



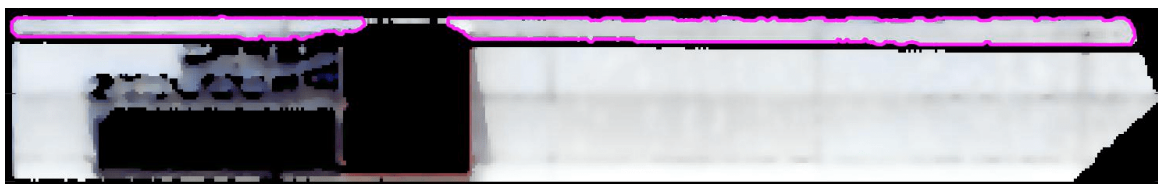
(a)



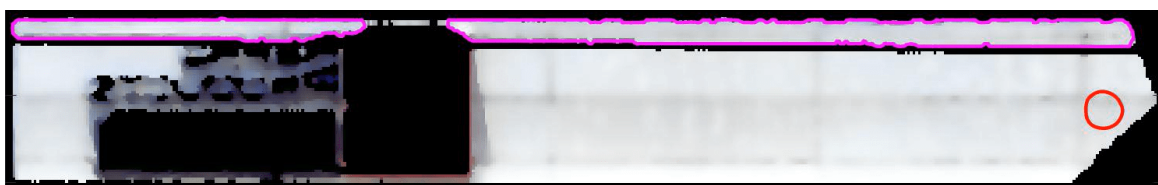
(b)



(c)



(d)



(e)

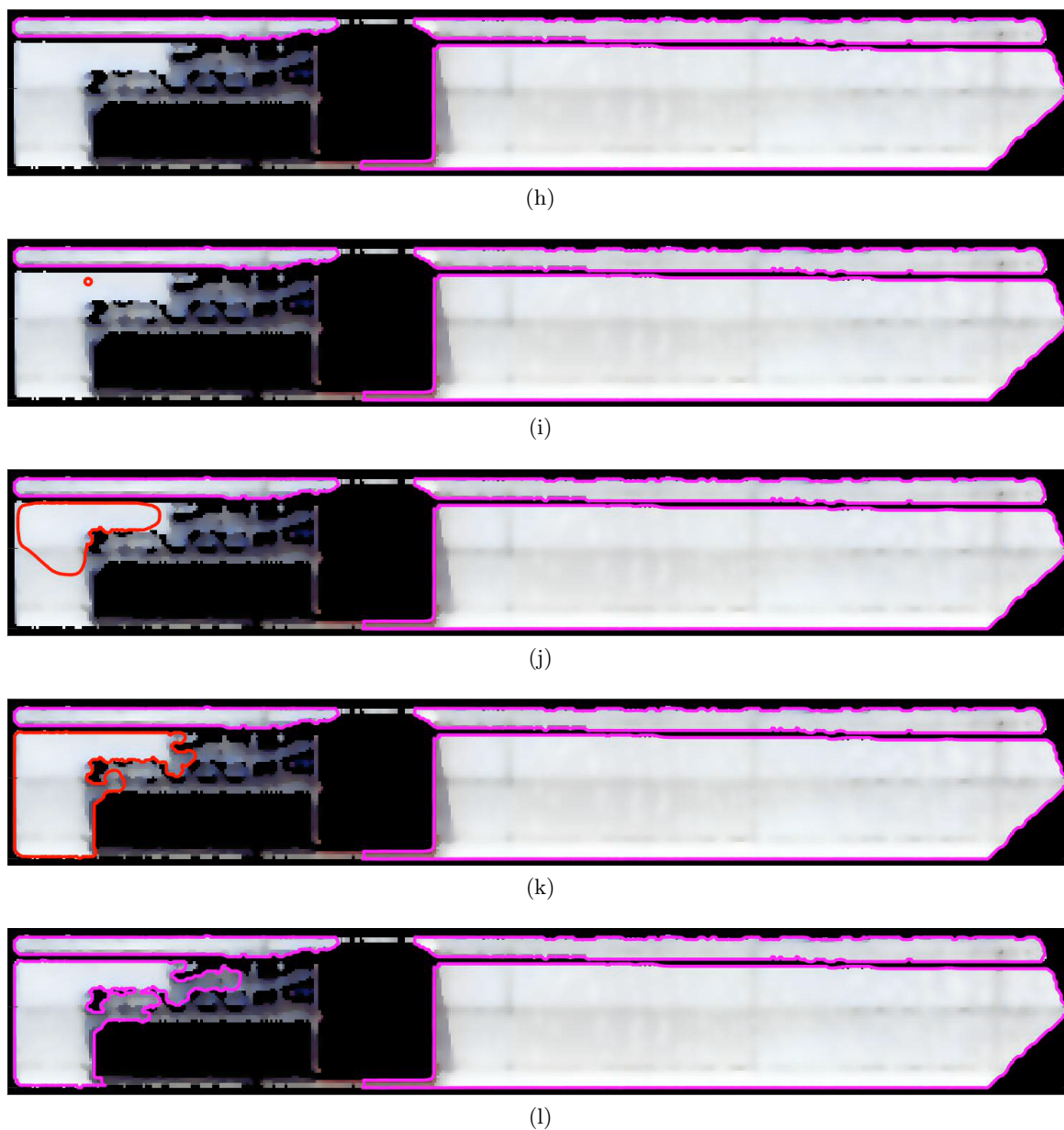


(f)



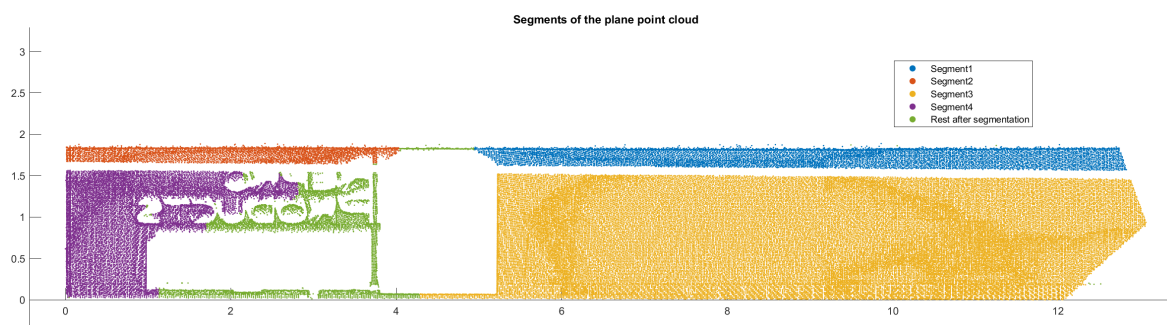
(g)

Obr. 4.20: Vývoj kriviek.



Obr. 4.20: Vývoj kriviek.

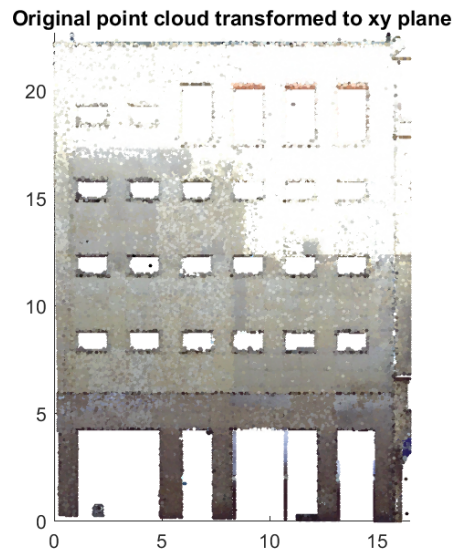
Nakoniec vytvoríme segmenty, ktoré môžeme vidieť na obrázku 4.21.



Obr. 4.21: Mračno bodov.

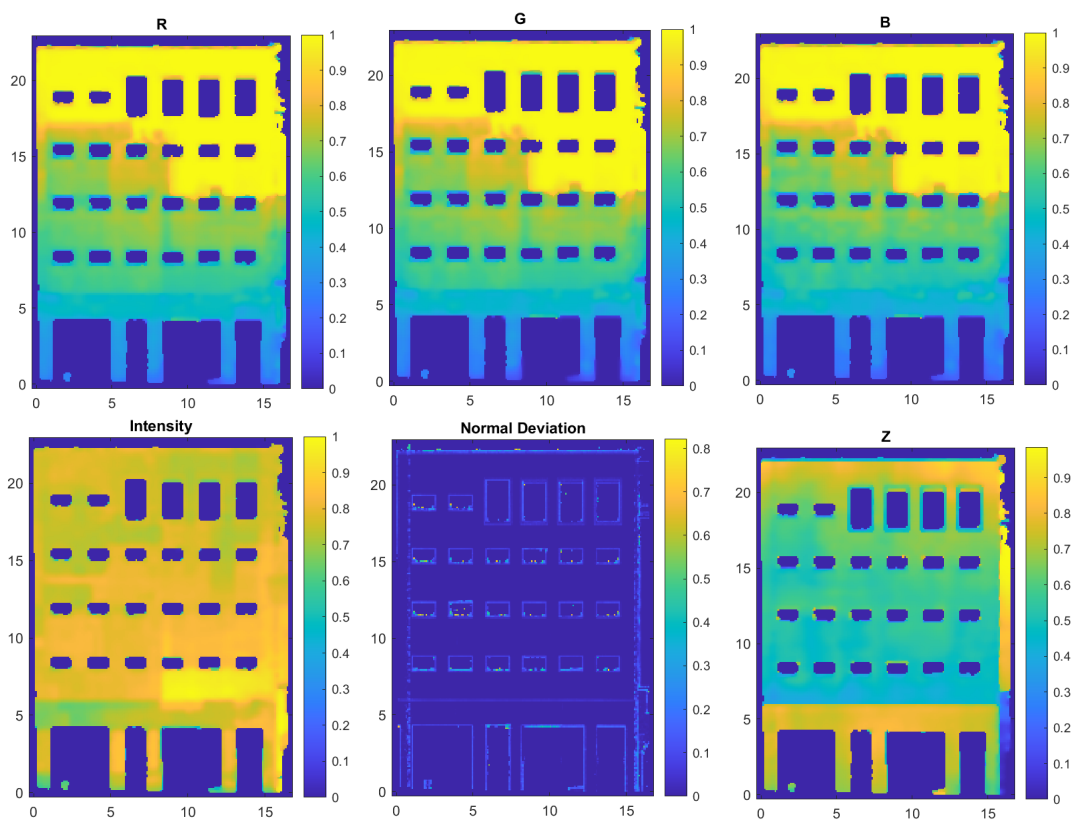
4.4 Poschodová budova

V tomto numerickom experimente budeme segmentovať mračno bodov, ktoré patrí časti fasády štvorposchodovej budovy Divadla P.O. Hviezdoslava. Mračno zafarbené tromi kanálmi R, G a B je na obrázku 4.22.

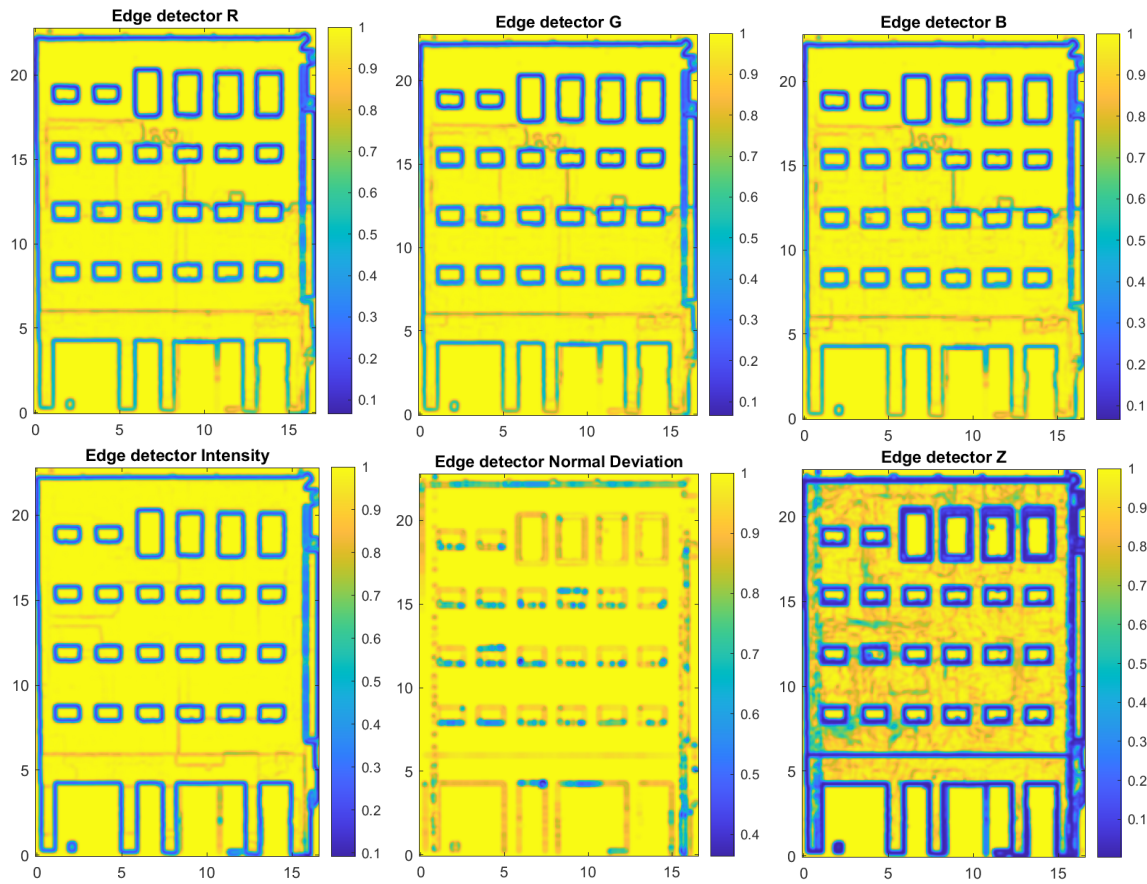


Obr. 4.22: Mračno bodov.

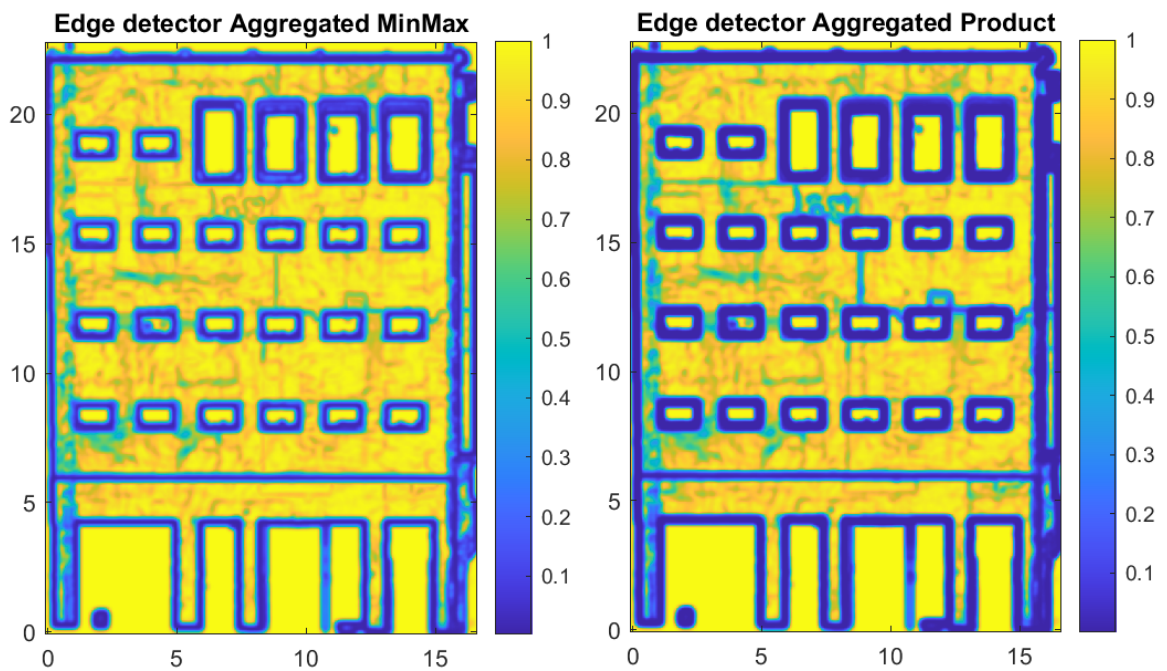
Opäť vypočítame pixelové reprezentácie pre vlastnosti mračna, ktoré vidíme na obrázku 4.23, ich hranové detektory sú na obrázku 4.23 a agregované hranové detektory na obrázku 4.24.



Obr. 4.23: Pixelová reprezentácia vlastností mračna bodov.

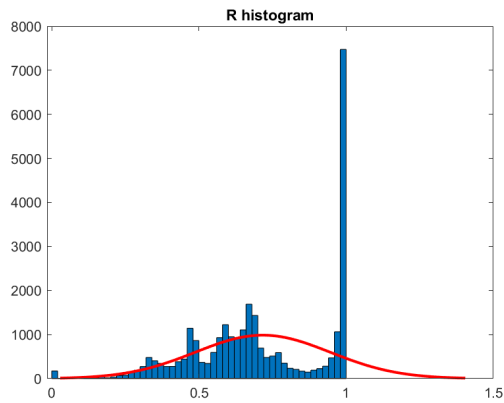
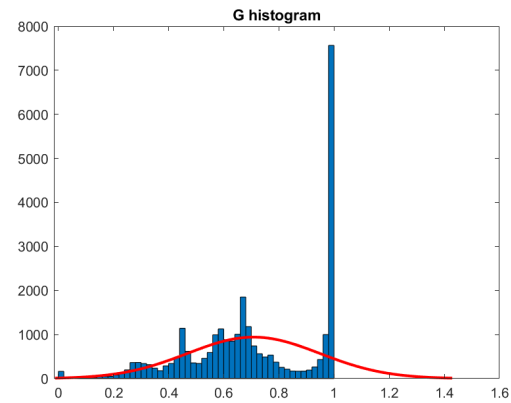
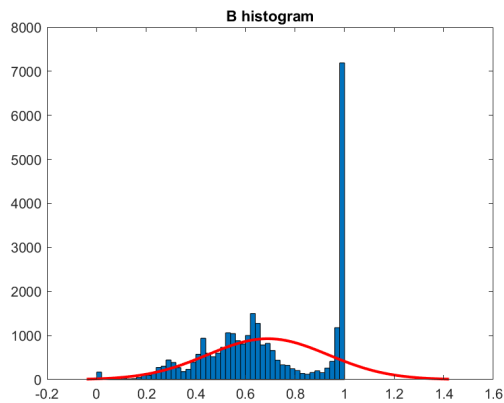
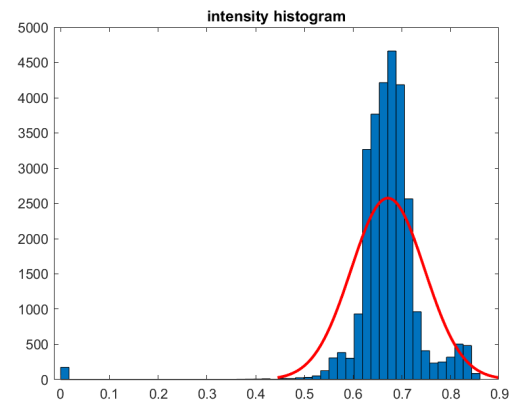
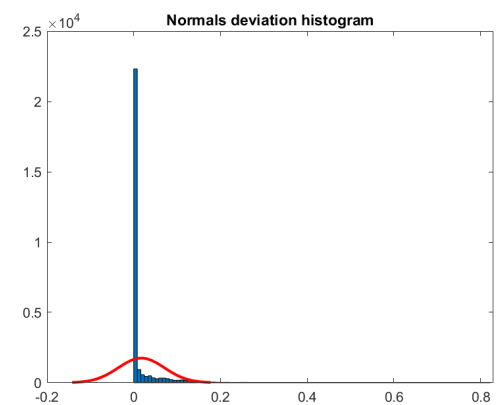
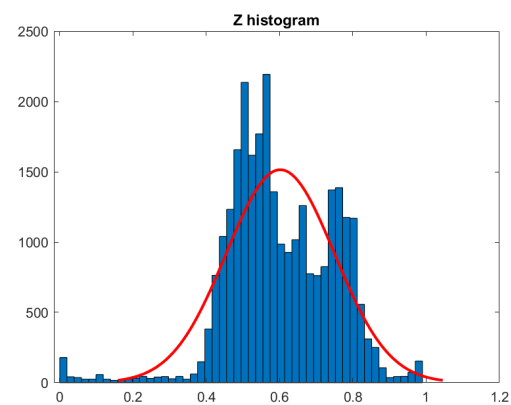


Obr. 4.23: Hranové detektory všech vlastností.



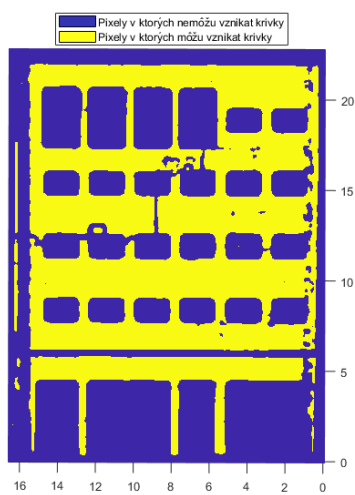
Obr. 4.24: Agregované hranové detektory.

Ďalej z histogramov vlastností vypočítame prahové hodnoty, ich vizualizáciu vidíme na obrázku 4.25.

(a) $\mu = 0.3017$ (b) $\mu = 0.3605$ (c) $\mu = 0.3620$ (d) $\mu = 0.3561$ (e) $\mu = 0.2728$ (f) $\mu = 0.0567$

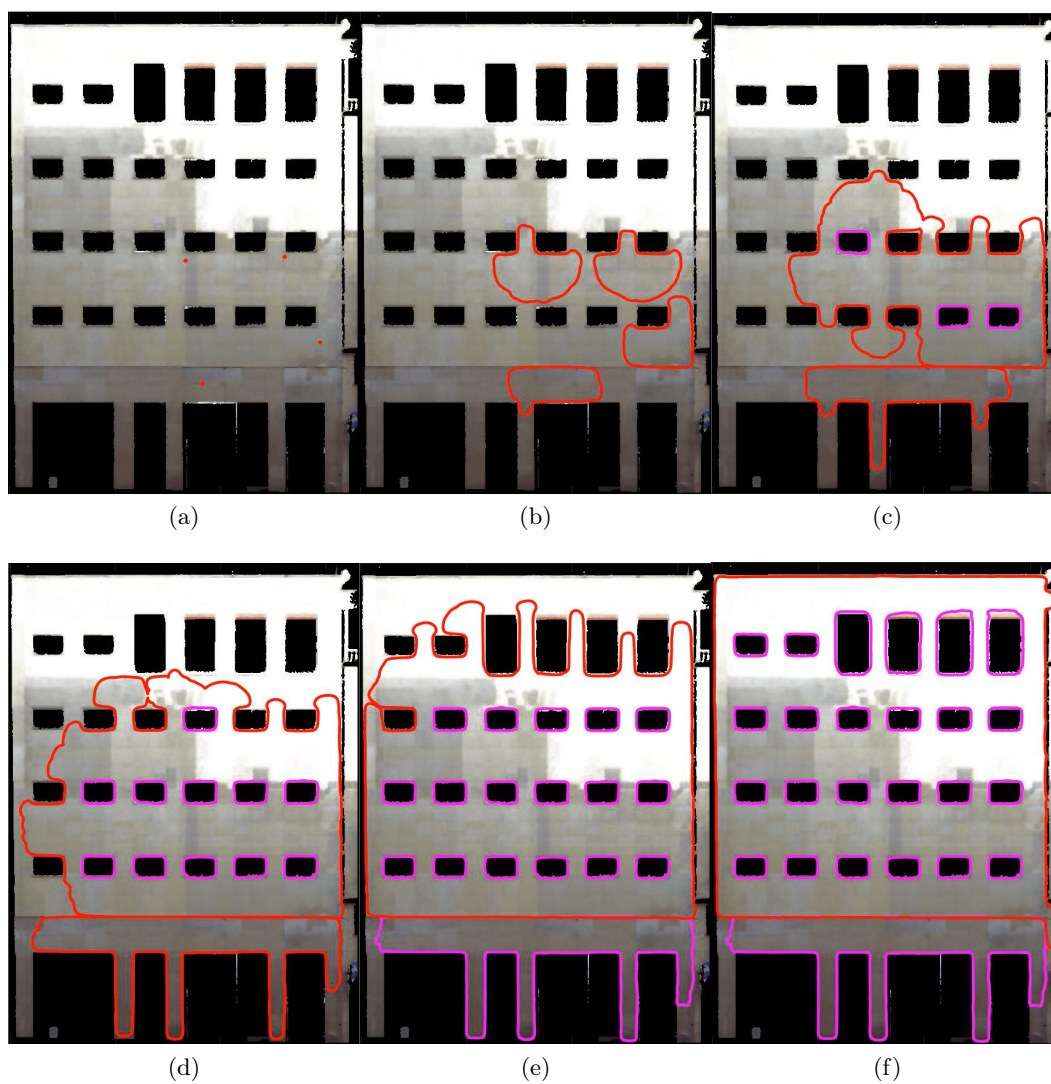
Obr. 4.25: Histogramy vlastností.

Posledným krokom pred tým ako vložíme počiatkové krivky je vypočítať pixely, ktoré spĺňajú podmienky na vloženie počiatkových kriviek. Vizualizáciu týchto pixelov je na obrázku 4.26.



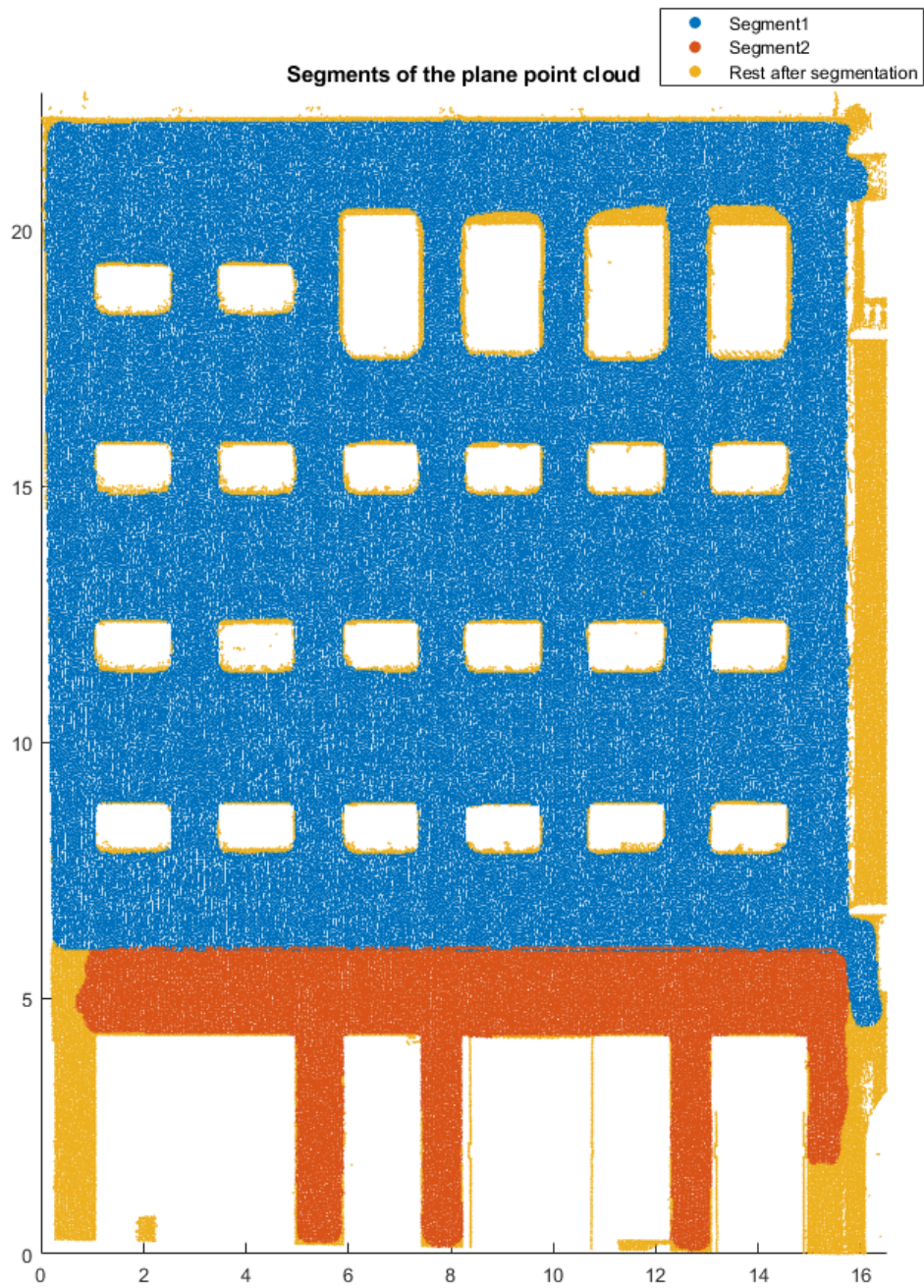
Obr. 4.26: Vizualizácia pixelov vhodných na vloženie počiatočných kriviek.

Vizualizáciu vývoja kriviek môžeme vidieť na obrázku 4.27.



Obr. 4.27: Vývoj kriviek.

Výsledkom segmentácie tejto regresnej roviny sú dva segmenty 4.28.



Obr. 4.28: Mračno bodov.

Kapitola 5

Záver

V tejto práci sme sa venovali segmentácii regresných rovín, ktoré obsahovali dáta vo forme mračna bodov. Tieto roviny predstavujú steny budov naskenované terestrickým laserovým skenerom, ktoré môžu obsahovať rôzne objekty ležiace v stenách.

Hlavným cieľom tejto práce bolo získať body mračna ležiace priamo v stene, a to použitím segmentácie vyvíjajúcimi sa krivkami. Na to sme vytvorili matematický model, ktorý popisujeme v kapitole 2, kde sme využili poznatky z oblasti diferenciálnej geometrie na vývoj kriviek. Tento model sme následne diskretizovali v kapitole 3, v ktorej opisujeme aj ďalšie nevyhnuté metódy a algoritmy potrebné na segmentáciu vyvíjajúcimi sa krivkami. Jednou z metód, ktorú sme úspešne aplikovali, bol algoritmus topologických zmien opísaný v článku [1], ktorý sa nám podarilo rozšíriť o kontrolu spájania kriviek na hranách medzi oblasťami s rôznymi priemernými hodnotami vlastností. Pomocou fuzzy logických operátorov zo sekcie 3.5, sme dokázali zapojiť všetky vlastnosti mračna bodov na vytvorenie agregovaného hranového detektora a funkcie homogenity, ktoré sú v našom modeli základom pre výpočet normálovej rýchlosti vyvíjajúcej sa krivky.

Všetky metódy popísané v kapitole 3 sme implementovali v programe MATLAB na vytvorenie algoritmu, ktorého postupnosť popisujeme v sekcii 3.9. Funkčnosť algoritmov sme demonštrovali na niekoľkých numerických experimentoch v kapitole 4.

Výsledné rozdelenie mračien bodov na niekoľko segmentov považujeme za uspokojivé. Algoritmus je schopný určiť kam vložiť počiatočné krivky, vyvíjať ich tak, aby segmentovali objekty ležiace v stene a dokázali sa spájať a rozdeľovať podľa toho, ako to je potrebné vzhľadom na vlastnosti mračna bodov a nakoniec rozdeliť mračno na segmenty.

Aj napriek tomu, že výsledky segmentácie sú uspokojivé stále je niekoľko oblastí, v ktorých je možnosť vylepšenia. Jednou z oblastí, ktorá by sa dala zlepšiť je návrh funkcie homogenity. Určenie prahovej hodnoty na základe histogramu môže byť skresľujúce, keďže v niektorých prípadoch je jasne vidieť už z histogramu, že segmentovaná stena obsahuje viac homogénnych oblastí. Takúto situáciu môžeme pozorovať na histogramoch pre R, G a B zložky farby z prvého numerického experimentu na obrázku 4.5. Na histogramoch zložiek B a G môžeme pozorovať, že vysoká početnosť pixelov je v okolí hodnôt 0.4 a 0.6, pričom medzi týmito hodnotami sa nachádza minimum. To môže značiť, že v stene ležia dva segmenty, čo potvrdzuje aj výsledok našej segmentácie, kde jeden segment tvorili body prislúchajúce stene a druhý body prislúchajúce dverám. V takomto prípade sa môže stať, že prahová hodnota môže byť nepresne určená a to by mohlo negatívne ovplyvniť vývoj krivky. Ďalej by sa dalo použiť fuzzy logiku pri overení podmienky spojenia kriviek na hranici dvoch oblastí, kde momentálne porovnávame priemery.

Literatúra

- [1] M. AMBROZ, M. BALAŽOVJECH, M. MEDL'A, AND K. MIKULA, *Numerical modeling of wildland surface fire propagation by evolving surface curves*, Advances in Computational Mathematics, 45 (2019).
- [2] M. BALAŽOVJECH, K. MIKULA, M. PETRÁŠOVÁ, AND J. URBÁN, *Lagrangean method with topological changes for numerical modelling of forest fire propagation*, Proceedings of the Conference Algoritmy, (2015), pp. 42–52.
- [3] A. KOPÁČIK, J. ERDÉLYI, P. KYRINOVÍČ, I. LIPTÁK, AND S. LUKÁČ, *Geodézia v priemysle*, (2016).
- [4] K. MIKULA, M. REMEŠÍKOVÁ, P. SARKOCI, AND D. ŠEVČOVIČ, *Manifold evolution with tangential redistribution of points*, SIAM Journal on Scientific Computing, 36 (2014), pp. A1384–A1414.
- [5] K. MIKULA, J. URBÁN, M. KOLLÁR, M. AMBROZ, I. JAROLÍMEK, J. SIBIK, AND M. ŠIBÍKOVÁ, *An automated segmentation of natura 2000 habitats from sentinel-2 optical data*, Discrete & Continuous Dynamical Systems - S, 14 (2021), pp. 1017–1032.
- [6] E. VAN KRIEKEN, E. ACAR, AND F. VAN HARMELEN, *Analyzing differentiable fuzzy logic operators*, CoRR, abs/2002.06100 (2020).