

SLOVENSKÁ TECHNICKÁ UNIVERZITA V BRATISLAVE
STAVEBNÁ FAKULTA

Evidenčné číslo: SvF-16645-115286

SEGMENTÁCIA OBLASTÍ SVETELNÉHO ZNEČISTENIA ZO
SATELITNÝCH SNÍMOK

Bakalárska práca

2024

Slávka Martinčeková

SLOVENSKÁ TECHNICKÁ UNIVERZITA V BRATISLAVE
STAVEBNÁ FAKULTA

SEGMENTÁCIA OBLASTÍ SVETELNÉHO ZNEČISTENIA ZO
SATELITNÝCH SNÍMOK

Bakalárska práca

Študijný program: Matematicko-počítačové modelovanie
Študijný odbor: 9.1.9. Aplikovaná matematika
Školiace pracovisko: Katedra matematiky a deskriptívnej geometrie
Školiteľ: Ing. Michal Kollár, PhD.

Bratislava 2024

Slávka Martinčeková



ZADANIE BAKALÁRSKEJ PRÁCE

Študentka: **Slávka Martinčeková**
ID študenta: 115286
Študijný program: matematicko-počítačové modelovanie
Študijný odbor: matematika
Vedúci práce: Ing. Michal Kollár, PhD.
Vedúci pracoviska: Ing. Marek Macák, PhD.

Názov práce: **Segmentácia oblastí svetelného znečistenia zo satelitných snímok**

Jazyk, v ktorom sa práca vypracuje: slovenský jazyk

Špecifikácia zadania:

Práca sa zaoberá segmentáciou oblastí svetelného znečistenia zo satelitných snímok získaných družicou Suomi-NPP, ktorá zachytáva nočné vyžarovanie Zeme. Cieľom práce je oboznámiť sa a implementovať metódy spracovania obrazu potrebné pre správnu segmentáciu polygonálnych modelov miest na účely modelovania rozloženia jasu v zložitom prostredí. Každé mesto môže byť reprezentované viacerými samostatnými polygónmi, pričom každý polygón má zahrňovať pixely s približne rovnakými hodnotami jasu. Takéto polygony slúžia ako vstup do softvéru SkyGlow, ktorý modeluje rozloženie jasu na oblohe. Práca je realizovaná v spolupráci s Ústavom stavebníctva a architektúry SAV.

Termín odovzdania bakalárskej práce: 02. 05. 2024
Dátum schválenia zadania bakalárskej práce: 22. 03. 2024
Zadanie bakalárskej práce schválil: prof. RNDr. Karol Mikula, DrSc. – garant študijného programu

Čestné prehlásenie

Prehlasujem, že som túto záverečnú prácu vypracovala samostatne pod vedením vedúceho záverečnej práce, s použitím literatúry uvedenej v zozname použitej literatúry.

Bratislava 2. 5. 2024

Slávka Martinčeková

Podakovanie

Týmto sa chcem poďakovať vedúcemu práce, Ing. Michalovi Kollárovi, PhD., za odbornú pomoc, pravidelné konzultácie, trpezlivý prístup a ochotu, ktoré boli prínosom pri písaní tejto bakalárskej práce.

Bratislava 2. 5. 2024

Slávka Martinčeková

Abstrakt

Názov práce: Segmentácia oblastí svetelného znečistenia zo satelitných snímok

Abstrakt: Práca sa zaoberá segmentáciou oblastí svetelného znečistenia zo satelitných snímok získaných družicou Suomi-NPP, ktorá zachytáva nočné vyžarovanie Zeme. Cieľom práce je oboznámiť sa a implementovať metódy spracovania obrazu potrebné pre správnu segmentáciu polygonálnych modelov miest na účely modelovania rozloženia jasů v zložitom prostredí. Každé mesto môže byť reprezentované viacerými samostatnými polygónmi, pričom každý polygón má zahrňovať pixely s približne rovnakými hodnotami jasů. Takéto polygony slúžia ako vstup do softvéru SkyGlow simulator, ktorý modeluje rozloženie jasů na oblohe. Práca je realizovaná v spolupráci s Ústavom stavebníctva a architektúry SAV.

Kľúčové slová: difúzna filtrácia, krivostná filtrácia, orientované izočiarý, segmentácia

Abstract

Title: Segmentation of areas affected by light pollution from satellite images

Abstract: This thesis focuses on the segmentation of areas affected by light pollution from satellite images obtained by the Suomi-NPP satellite which captures the Earth's nighttime radiance. The aim of the work is to familiarize and implement image processing methods necessary for proper segmentation of polygonal models of cities for the purpose of modeling the distribution of brightness in complex environments. Each city may be represented by multiple separate polygons, with each polygon encompassing pixels with approximately the same brightness values. Such polygons serve as input to the SkyGlow Simulator software which models the distribution of brightness in the sky. The work is carried out in collaboration with the Institute of Construction and Architecture of the Slovak Academy of Sciences.

Keywords: diffuse filtering, curvature filtering, oriented isocurves, segmentation

Obsah

1	Úvod	9
2	Vstupné dáta	11
2.1	Konverzia do textového súboru	11
2.2	Interpolácia farieb	12
3	Filtrácia	15
3.1	Lineárna filtrácia pomocou rovnice vedenia tepla	15
3.1.1	Okrajová úloha s počiatočnou podmienkou	15
3.1.2	Časová diskretizácia	15
3.1.3	Priestorová diskretizácia	16
3.1.4	Explicitná schéma na riešenie rovnice vedenia tepla	17
3.1.5	Implicitná schéma na riešenie rovnice vedenia tepla	18
3.2	Geodesic mean curvature flow (GMCF)	19
3.2.1	Okrajová úloha s počiatočnou podmienkou	19
4	Izočiary	23
4.1	Algoritmus na detekciu izočiar	23
5	Numerické experimenty	27
5.1	Filtrácia lineárnou difúziou	27
5.2	Filtrácia pomocou GMCF	28
5.3	Izočiary	31
6	Záver	35
	Bibliografia	37

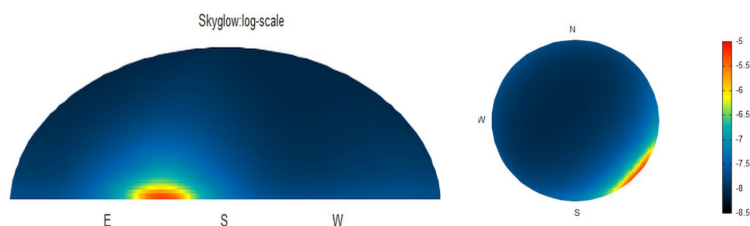
Kapitola 1

Úvod

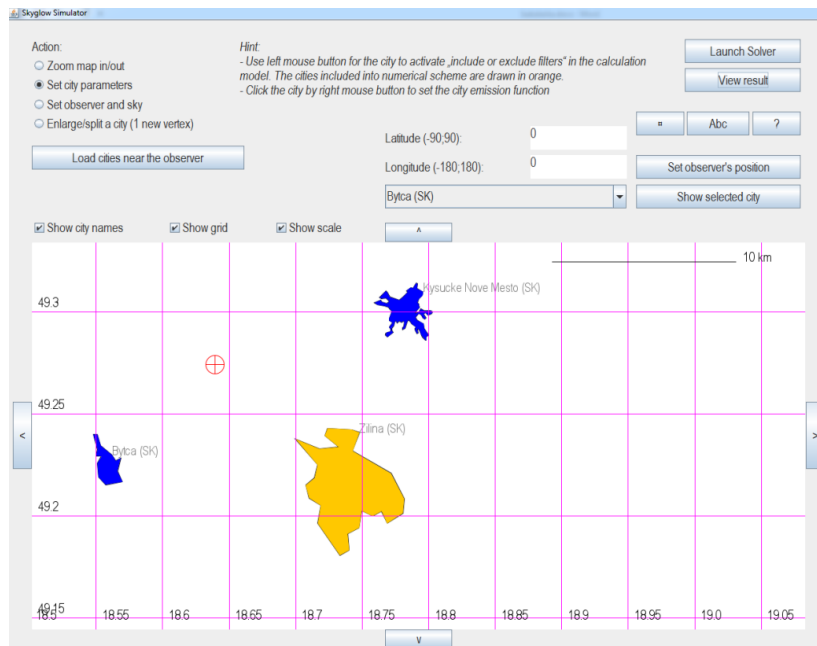
Modelovanie rozloženia jasu na nočnej oblohe a svetelného znečistenia má široké spektrum využitia v astronómii, urbanistike, mestskom plánovaní a ochrane životného prostredia. Je užitočné na identifikovanie miest s minimálnym znečistením pre pozorovanie vesmírnych objektov alebo naopak miest s vysokým znečistením, kde je vhodné navrhnúť opatrenia na zníženie svetelného znečistenia a tým chrániť životné prostredie. Pri metskcom plánovaní sa využíva model svetelného znečistenia na optimalizáciu verejného osvetlenia a identifikáciu oblastí s nadmerným alebo nedostatočným osvetlením.

Cieľom práce je vytvoriť aplikáciu s užívateľským prostredím, implemetnovanú v jazyku C++, s použitím knižníc Qt [13]. Načítajú do nej satelitné snímky nočného svetelného žiarenia vo formáte ASCII Grid a prefiltrujú sa pomocou modelov lineárnej difúzie, Mean curvature flow alebo Geodesic mean curvature flow. Po filtrácii sa vysegmentujú oblasti na základe ich svetelného žiarenia, ktoré predstavujú mestá a nájdu sa ich hranice vo forme polygónov. Polygonálne modely miest slúžia ako vstup do nástroja SkyGlow simulator [2], vyvinutej na Ústave stavebníctva a Architektúry Slovenskej akadémie vied a Fakulte matematiky, fyziky a informatiky Univerzity Komenského v Bratislave. Užívateľské prostredie tohto nástroja je na obrázku Obr. 1.2, kde sú zobrazené 3 modely miest, pričom mesto zobrazené žltým polygónom je aktívne a pozícia pozorovateľa je znázornená červeným krížkom. Výstupom je model rozloženia jasu na oblohe v dvoch typoch zobrazení (Obr. 1.1), kde aktívne mesto pôsobí ako zdroj svetelného žiarenia. Polygonálne modely miest, ktoré sa v tomto nástroji už nachádzajú, boli vysegmentované ručne a našim cieľom je tento proces zautomatizovať.

Teoretická časť práce sa nachádza v kapitolách 2, 3 a 4. V kapitole 2 sa zaoberáme vstupnými dátami do aplikácie, ich spracovaním a vizualizáciou. V kapitole 3 odvodíme explicitnú (3.16) a implicitnú (3.25) numerickú schému pre výpočet lineárnej difúzie, a taktiež semi-implicitnú schému (3.39) pre výpočet modelu Geodesic mean curvature flow. V kapitole 4 predstavíme algoritmus na detekciu izočiar, ktorý segmentuje obraz pomocou prahovej hodnoty a následne nájde hranice regiónov reprezentované polygónmi. Praktická časť práce je v kapitole 5, kde prezentujeme výsledky našich numerických experimentov.



Obr. 1.1: Model rozloženia jasu na oblohe



Obr. 1.2: Uživatelské prostredie Skyglow simulator

Kapitola 2

Vstupné dáta

Originálne dáta pochádzajú z databázy VIIRS, ktorej dáta sú získavané zo senzorov VIIRS umiestnených na družiciach. Senzor VIIRS (Visible Infrared Imaging Radiometer Suite) je družicový senzor, ktorý zhromažďuje snímky a rádiometrické merania pevniny, atmosféry, kryosféry a oceánov vo viditeľnom a infračervenom pásme elektromagnetického spektra, pričom skenuje pás široký približne 3040 km [6].

Naše dáta boli nasnímané družicou Suomi NPP (Suomi National Polar-orbiting Partnership). Je to meteorologická družica prevádzkovaná Národným úradom pre letectvo a vesmír (NASA) a Úradom pre atmosféru a oceány (NOAA). Bola vypustená na obežnú dráhu v októbri 2011 a v súčasnosti je stále v prevádzke. Táto družica je na obežnej dráhe so Slnkom synchronizovanou polárno-orbitálnou dráhou, čiže krúži okolo Zeme v blízkosti polárnych regiónov. Poskytuje pokrytie celej Zeme každých 12 hodín (dvakrát denne) a prechádza cez rovnaké miesta za svetla aj tmy, približne v rovnakom čase pri každom prechode [8]. Je pomenovaná podľa fínsko-amerického meteorológa Vernera E. Suomiho. Poskytuje dôležité údaje pre predpovedanie počasia, výskum klímy a monitorovanie životného prostredia [5].

DNB žiarenie pochádza zo senzora Day-Night Band (DNB), ktorý je súčasťou senzora VIIRS. Jednou z jeho významných vlastností je veľmi citlivá detekcia osvetlenia prítomného na zemskom povrchu generovaného elektrickou energiou. Vďaka tomu je vhodný na detekciu svetelného znečistenia a monitorovanie nočného osvetlenia [7].

Dáta zo senzora VIIRS sú uložené vo formáte HDF-5 (Hierarchical Data Format version 5). Je to štruktúrovaný a flexibilný formát na ukladanie a organizovanie veľkých množstiev dát. Je to pokročilá verzia formátu HDF, ktorá ponúka vylepšenú podporu pre rôzne typy dát a lepšiu efektívnosť pri ukladaní a manipulácii s veľkými objemami údajov [12].

2.1 Konverzia do textového súboru

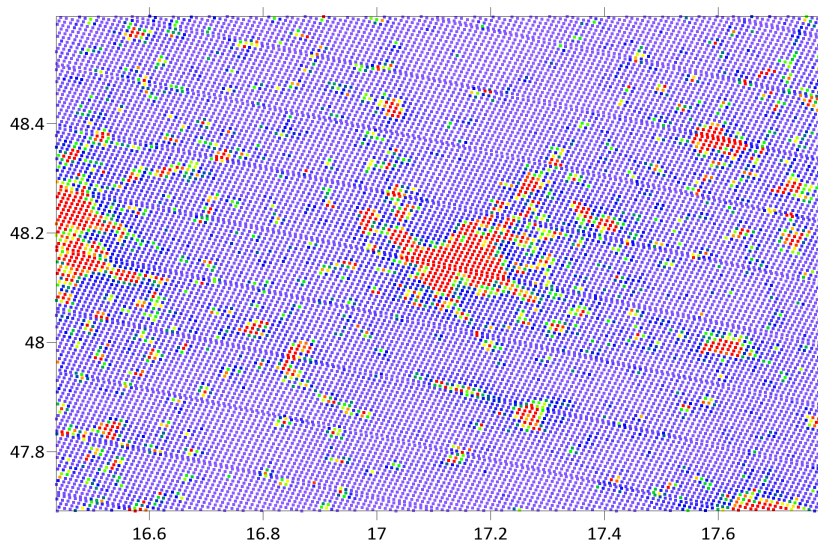
Dáta uložené v textovom súbore nám boli poskytnuté vďaka spolupráci s Ústavom stavebníctva a Architektúry Slovenskej akadémie vied. Súbor obsahuje 4 kanály extrahované z formátu HDF-5, popísané v tabuľke 2.1 [9]. Pracujeme s tromi testovacími súborami, ktoré zahŕňujú plochu $100 \text{ km} \times 100 \text{ km}$, pričom v strede sú mestá Bratislava, Banská Bystrica a Košice.

Tabuľka 2.1: Kanály v textovom súbore

Veličina	Jednotky
DNB žiarenie	$Wcm^{-2}sr^{-1}$
zemepisná šírka	stupne
zemepisná dĺžka	stupne
zenitný uhol	stupne

Zenitný uhol má vplyv na žiaru pixela, keďže žiara závisí od uhla, pod ktorým družica sníma daný pixel. V rámci jedného mesta je však vplyv zenitného uhla na žiaru zanedbateľný, preto ho nebudeme ďalej uvažovať. Keďže hodnoty intenzity DNB žiarenia sú veľmi malé, tento stĺpec vynásobíme číslom 10^{10} a tým multiplikatívne preškálujeme satelitnú snímku.

Grafické zobrazenie dátového súboru obsahuje nepravidelnú mriežku (Obr. 2.1), preto sme dáta interpolovali do pravidelnej mriežky metódou najbližšieho suseda pomocou softvéru Surfer [11]. Krok mriežky sme nastavili podľa priemernej vzdialenosti bodov. Výsledkom je súbor vo formáte ASCII grid, ktorý obsahuje metadáta o rozsahu pravidelnej mriežky a hodnoty intenzít svetelného žiarenia v ASCII formáte.



Obr. 2.1: Nepravidelná mriežka pôvodných dát

2.2 Interpolácia farieb

Na vizualizáciu intenzity DNB žiarenia použijeme funkciu prenosu farby, ktorá každej intenzite priradí farbu. Vstupom do funkcie prenosu farby sú hodnoty intenzít v ich pôvodnom rozsahu.

Máme definovanú farebnú paletu s n farbami $\{f_1, \dots, f_n\}$. Každá farba má svoju pozíciu vo farebnej palete p_i , pričom $p_i \in [0, 1]$ a $p_i \neq p_j$, kde $i, j = 1, \dots, n$ a $i \neq j$. Intenzitu I prispôbíme rozsahu intenzít $[I_{min}, I_{max}]$ vzťahom

$$c = \frac{I - I_{min}}{I_{max} - I_{min}}, \quad (2.1)$$

kde I_{min} a I_{max} sú hodnoty zadané užívateľom, pričom $I_{min} < I_{max}$. Ak hodnoty I_{min} a I_{max} predstavujú skutočné minimum a maximum intenzít dátového súboru, potom vzorec (2.1) predstavuje normalizáciu intenzít do rozsahu $[0, 1]$. Prispôbenie hodnôt do rozsahu intenzít vykonávme len za účelom vizualizácie dát a nemeníme tým pôvodný dátový súbor.

Ak $p_i \leq c < p_{i+1}$, $i = 1, \dots, n - 1$, výsledná farba vznikne lineárnou interpoláciou medzi farbami f_i a f_{i+1} . Ak $c < 0$ výsledná farba bude f_1 , a ak $c \geq 1$ výsledná farba bude f_n . Hodnoty intenzít menšie ako I_{min} a väčšie ako I_{max} budú pri vykreslení orezané.

Použijeme farebné palety z obrázku Obr. 2.2 a vizualizujeme tri testovacie súbory na obrázkoch Obr. 2.3 a Obr. 2.4. Pôvodné rozsahy intenzít sú pre snímku Bratislava $[1.36, 1608.49]$, Banská Bystrica $[0.02, 450.12]$ a Košice $[0.09, 2720.48]$. Rozsah intenzít pre vizualizáciu snímok sme zvolili $[0, 100]$.

Algorithm 1: Funkcia prenosu farby

Data: intenzita I , minimálna intenzita I_{min} , maximálna intenzita I_{max} , farebná paleta pal

Result: farba

$diff \leftarrow I_{max} - I_{min}$;

if $diff = 0$ **then**

 | **return** f_n ;

end

$c \leftarrow (I - I_{min})/diff$;

if $c \geq 1$ **then**

 | **return** f_n ;

end

for $i = 2$ **to** n **do**

 | **if** $c < p_i$ **then**

 | $c \leftarrow (c - p_{i-1})/(p_i - p_{i-1})$;

 | **return** $(1 - c) * f_{i-1} + c * f_i$

 | **end**

end

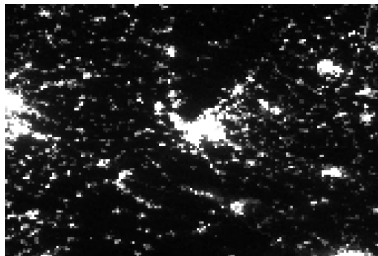


(a) Paleta Grayscale

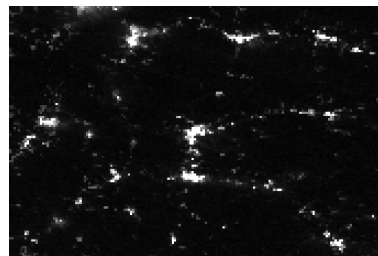


(b) Paleta Rainbow

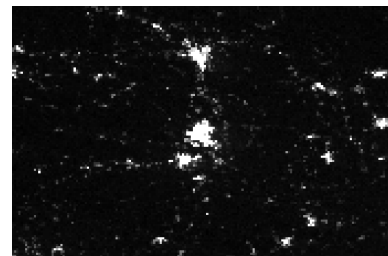
Obr. 2.2: Ukážka farebných paliet



(a) Bratislava

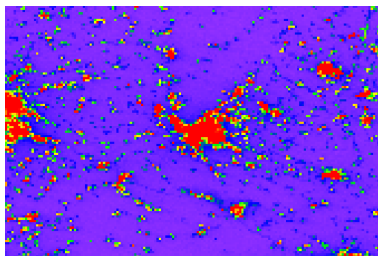


(b) Banská Bystrica

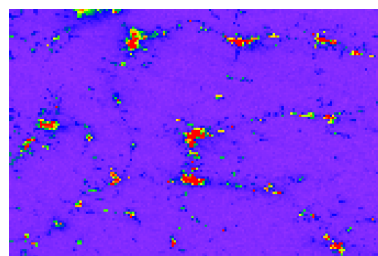


(c) Košice

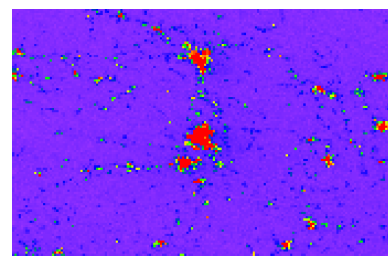
Obr. 2.3: Vizualizácia DNB žiarenia s farebnou paletou Grayscale a rozsahom intenzít [0,100]



(a) Bratislava



(b) Banská Bystrica



(c) Košice

Obr. 2.4: Vizualizácia DNB žiarenia s farebnou paletou Rainbow a rozsahom intenzít [0,100]

Kapitola 3

Filtrácia

Na satelitných snímkach sa náchádza šum, na ktorý môžu vplývať rôzne faktory, ako napríklad atmosférické podmienky alebo technické obmedzenia senzorov. Pre získanie čo najpresnejších výsledkov je potrebné tento šum eliminovať. Na snímkach sa tiež nachádzajú veľmi malé oblasti, veľkosti niekoľkých pixelov, ktoré nemusia mať zásadný vplyv na model rozloženia jasů na oblohe. Preto nemusí byť potrebné z nich vytvárať polygóny, o čom rozhoduje užívateľ výberom parametrov modelu.

Obe tieto požiadavky vyriešime filtráciou obrazu. Lineárna filtrácia založená na rovnici vedenia tepla zhladzuje rovnomerne a použijeme ju na predhľadanie. Nelineárna filtrácia založená na modeli Geodesic mean curvature flow zhladzuje adaptívne, zachováva hrany a hlavné štruktúry, vytvára homogénne oblasti s určitou úrovňou jasů a pritom odstraňuje šum.

3.1 Lineárna filtrácia pomocou rovnice vedenia tepla

Rovnica vedenia tepla (lineárna difúzia) je odvodená zo zákona zachovania energie a empirického Fourierovho zákona. Princíp minimalizácie energie, ktorý je základom rovnice vedenia tepla, hovorí, že systém sa snaží dosiahnuť energeticky najvýhodnejšiu konfiguráciu. Preto táto rovnica je rovnicou najmenšieho povrchu, teda hodnota v danom bode je určená ako priemer hodnôt z jeho okolia a minimalizuje gradienty, čo sa v našej aplikácii prejaví ako zhladzovanie obrazu.

3.1.1 Okrajová úloha s počiatočnou podmienkou

Okrajová úloha s počiatočnou podmienkou pre rovnicu vedenia tepla s nulovou Neumanovou okrajovou podmienkou má tvar

$$\frac{\partial u}{\partial t} = \Delta u, \quad x \in \Omega, \quad t \in [0, \sigma], \quad (3.1)$$

$$\frac{\partial u}{\partial \vec{n}} = 0, \quad x \in \partial\Omega, \quad t \in [0, \sigma], \quad (3.2)$$

$$u(x, 0) = u^0(x), \quad x \in \Omega, \quad (3.3)$$

kde $u(x, t)$ je intenzita, \vec{n} je jednotková vonkajšia normála ku hranici $\partial\Omega$ a $u^0(x)$ je počiatočný obraz [3].

3.1.2 Časová diskretizácia

Rovnicu (3.1) diskretizujeme v čase pomocou konečných diferencií. Najskôr si zvolíme časový krok τ a podelíme ním časový interval $[0, \sigma]$ na N časových úsekov. Numerické riešenie v n -tom časovom kroku budeme označovať u^n . Ďalej nahradíme parciálnu deriváciu podľa času

konečnou diferenciou. Môžeme použiť doprednú alebo spätnú diferenciu. Dopredná diferenciacia má tvar

$$\frac{u^{n+1} - u^n}{\tau} \quad (3.4)$$

a spätná diferenciacia má tvar

$$\frac{u^n - u^{n-1}}{\tau}. \quad (3.5)$$

Pri doprednej diferencii je neznáma u^{n+1} a pri spätnej je to u^n . Centrálnu diferenciu nemôžeme použiť, pretože na výpočet riešenia u^n by sme potrebovali poznať riešenie u^{n+1} . Po dosadení do rovnice (3.1) dostaneme dve časové diskretizácie.

Explicitná časová diskretizácia

$$\frac{u^{n+1} - u^n}{\tau} = \Delta u^n = \nabla \cdot (\nabla u^n) \quad (3.6)$$

Nahradili sme parciálnu deriváciu podľa času doprednou diferenciou a pravú stranu uvažujeme v starom časovom kroku u^n . Nazýva sa explicitná, lebo neznámu u^n vieme vyjadriť v explicitnom tvare.

Implicitná časová diskretizácia

$$\frac{u^n - u^{n-1}}{\tau} = \Delta u^n = \nabla \cdot (\nabla u^n) \quad (3.7)$$

V tomto prípade sme použili spätnú diferenciu a pravú stranu uvažujeme v novom časovom kroku u^n . V tejto rovnici je neznáma u^n a jej riešenie získame v ďalších krokoch riešením sústavy lineárnych rovníc.

3.1.3 Priestorová diskretizácia

Na priestorovú diskretizáciu využijeme metódu konečných objemov. V tejto metóde je riešenie vo vnútri jednej bunky aproximované ako jedna hodnota. Táto hodnota predstavuje priemer hodnôt v danom objeme. Metóda konečných objemov má vlastnosť konzervativity, ktorá je zabezpečená integrovaním rovnice cez jednotlivé bunky, vďaka čomu bude tok cez hranice zahrnutý do výpočtu a bilancia tokov bude automaticky splnená.

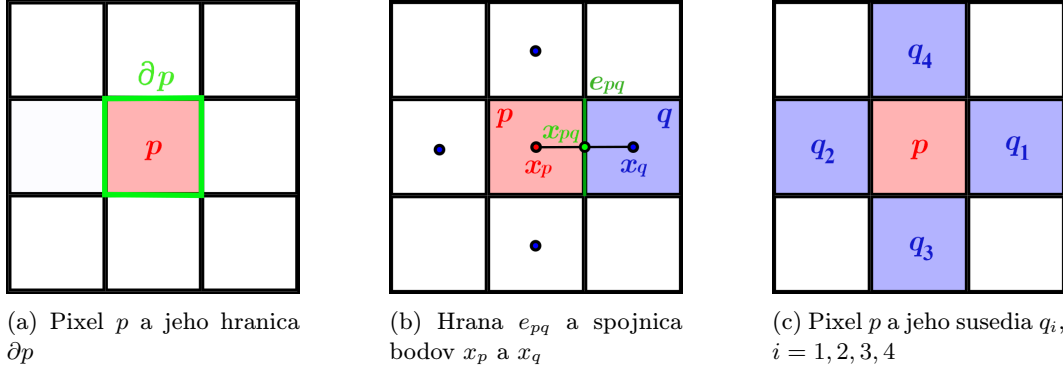
Za konečný objem uvažujeme pixel, konečno objemová sieť je množina pixelov, v našom prípade je to mapa intenzít.

Zavedenie označenia

Konečný objem budeme uvažovať ako otvorenú množinu a budeme ju označovať p . Hranicu konečného objemu budeme označovať ∂p (Obr. 3.1a). Uzáver konečného objemu označíme \bar{p} a platí $\bar{p} = p \cup \partial p$. Dĺžku strany konečného objemu označíme h a mieru konečného objemu $m(p)$, pričom v dvojrozmernom prípade ide o plochu.

Množinu pixelov označíme T_h . Spoločnú hranu dvoch buniek p a q označíme e_{pq} , teda $e^{pq} = \bar{p} \cap \bar{q}$ (Obr. 3.1b). Jej mieru označíme $m(e_{pq})$, pre dvojrozmernú bunku ide o dĺžku. Množinu všetkých dvojíc susedných konečných objemov označíme ϵ , teda $\epsilon = \{(p, q) \in T_h, \text{ kde } p \neq q \text{ a } m(e_{pq}) \neq 0\}$. Pre každú bunku p označíme $N(p)$ množinu jej susedov, s ktorými má spoločnú hranicu nenulovej miery, teda $N(p) = \{p \in \epsilon, \text{ takých, že } (p, q) \in \epsilon\}$ (Obr. 3.1c).

Vektor jednotkovej vonkajšej normály ku hranici pixela p označíme \vec{n}_p . Ak je tento vektor orientovaný z p do q označíme ho \vec{n}_{pq} . Normálu výpočítame ako $\vec{n}_{pq} = \frac{x_q - x_p}{|x_q - x_p|}$, kde x_q a x_p sú stredy pixelov q a p . Spojnica bodov x_q a x_p je kolmá na hranu e_{pq} . Dĺžku tejto spojnice označíme $d_{pq} = |x_q - x_p|$ a x_{pq} označíme priesečník tejto spojnice s hranou e_{pq} .



Obr. 3.1: Priestorová diskretizácia

3.1.4 Explicitná schéma na riešenie rovnice vedenia tepla

Budeme vychádzať z rovnice (3.6). Najskôr ju zintegrujeme nad konečným objemom p a dostaneme

$$\int_p \frac{u^{n+1} - u^n}{\tau} dx = \int_p \nabla \cdot (\nabla u^n) dx. \quad (3.8)$$

Na pravej strane použijeme Gaussovú vetu (iné názvy Gaussova-Ostrogradského, Greenova, divergenčná veta) a máme

$$\int_p \frac{u^{n+1} - u^n}{\tau} dx = \int_{\partial p} \nabla u^n \cdot \vec{n}_p dS. \quad (3.9)$$

Integrál po hranici konečného objemu na pravej strane môžeme vzhľadom na jeho geometriu vyjadriť ako súčet integrálov po jeho hranách. Dostaneme slabú konečno objemovú formuláciu úlohy

$$\int_p \frac{u^{n+1} - u^n}{\tau} dx = \sum_{q \in N(p)} \int_{e_{pq}} \nabla u^n \cdot \vec{n}_{pq} dS = \sum_{q \in N(p)} \int_{e_{pq}} \frac{\partial u^n}{\partial \vec{n}_{pq}} dS. \quad (3.10)$$

Z metódy konečných objemov vieme, že približné riešenie na jednom konečnom objeme je konštantné, a ak reprezentatívnu hodnotu numerického riešenia v čase n na pixeli p označíme u_p^n , ľavú stranu rovnice (3.10) môžeme napísať v tvare

$$\int_p \frac{u^{n+1} - u^n}{\tau} dx = \frac{u_p^{n+1} - u_p^n}{\tau} \int_p 1 dx = \frac{u_p^{n+1} - u_p^n}{\tau} m(p). \quad (3.11)$$

Člen $\nabla u^n \cdot \vec{n}_p$ aproximujeme na hrane e_{pq} konečnou diferenciou v bode x_{pq}

$$\nabla u^n \cdot \vec{n}_p = \frac{\partial u^n}{\partial \vec{n}_{pq}} \approx \frac{u_q^n - u_p^n}{d_{pq}}. \quad (3.12)$$

Potom pravú stranu rovnice (3.10) môžeme aproximovať

$$\sum_{q \in N(p)} \int_{e_{pq}} \nabla u^n \cdot \vec{n}_{pq} dS \approx \sum_{q \in N(p)} m(e_{pq}) \frac{u_q^n - u_p^n}{d_{pq}}. \quad (3.13)$$

Dostávame explicitnú schému pre všeobecnú sieť v tvare

$$m(p) \frac{u_p^{n+1} - u_p^n}{\tau} = \sum_{q \in N(p)} m(e_{pq}) \frac{u_q^n - u_p^n}{d_{pq}}, \quad (3.14)$$

kde

$$u_p^0 = \frac{1}{m(p)} \int_p u^0(x) dx, \quad p \in T_h. \quad (3.15)$$

Po úpravách dostaneme schému v tvare

$$u_p^{n+1} = u_p^n + \frac{\tau}{m(p)} \sum_{q \in N(p)} m(e_{pq}) \frac{u_q^n - u_p^n}{d_{pq}}. \quad (3.16)$$

Sieť pixelov je rovnomerná, hrana pixela má dĺžku 1, rovnako ako aj plocha pixela. Pre náš model teda platí $m(e_{pq}) = d_{pq} = 1$ a $m(p) = 1$. Tieto vzťahy dosadíme do (3.16) a upravíme, dostaneme explicitnú schému na riešenie rovnice vedenia tepla pre sieť pixelov

$$u_p^{n+1} = \left(1 - \tau \sum_{q \in N(p)} 1 \right) u_p^n + \tau \sum_{q \in N(p)} u_q^n. \quad (3.17)$$

Nulovú Neumannovu podmienku zabezpečíme tým, že člen $u_q^n - u_p^n$ v rovnici (3.16) je rovný nule, ak p je bunka na hranici siete pixelov a q je bunka mimo siete pixelov.

Schéma (3.17) je stabilná, ak pre koeficienty $1 - \tau \sum_{q \in N(p)} 1$ a τ platí, že sú z intervalu $[0, 1]$ a ich súčet je rovný 1. Súčet koeficientov je rovný 1 pre ľubovoľnú voľbu τ . Podmienka $1 - \tau \sum_{q \in N(p)} 1 \leq 1$ platí pre ľubovoľnú voľbu τ . Ďalej musí platiť $1 - \tau \sum_{q \in N(p)} 1 \geq 0$, čo upravíme a dostaneme

$$1 - \tau \sum_{q \in N(p)} 1 \geq 0, \quad (3.18)$$

$$1 - \tau \text{card}(N(p)) \geq 0, \quad (3.19)$$

$$\tau \text{card}(N(p)) \leq 1, \quad (3.20)$$

$$\tau \leq \frac{1}{\text{card}(N(p))}, \quad (3.21)$$

kde $\text{card}(N(p))$ reprezentuje kardinalitu množiny $N(p)$, teda počet jej prvkov. Pre sieť pixelov platí $\text{card}(N(p)) = 4$. Po dosadení získame podmienku stability explicitnej schémy 3.17 v tvare

$$\tau \leq \frac{1}{4}. \quad (3.22)$$

Touto podmienkou automaticky zabezpečíme aj $\tau \leq 1$ [3].

3.1.5 Implicitná schéma na riešenie rovnice vedenia tepla

Pri implicitnej schéme vychádzme z rovnice (3.7) a pri priestorovej diskretizácii budeme postupovať rovnako ako pri explicitnej schéme. Dostaneme schému v tvare

$$\frac{m(p)}{\tau} (u_p^n - u_p^{n-1}) = \sum_{q \in N(p)} m(e_{pq}) \frac{u_q^n - u_p^n}{d_{pq}}. \quad (3.23)$$

Dosadíme do nej vzťahy pre sieť pixelov $m(e_{pq}) = d_{pq} = 1$ a $m(p) = 1$, upravíme a dostaneme lineárny systém rovníc

$$u_p^n + \tau \sum_{q \in N(p)} (u_p^n - u_q^n) = u_p^{n-1} \quad (3.24)$$

a po úprave dostaneme

$$(1 + \tau \sum_{q \in N(p)} 1) u_p^n - \tau \sum_{q \in N(p)} u_q^n = u_p^{n-1}. \quad (3.25)$$

Implicitná schéma je bezpodmienečne stabilná vzhľadom na veľkosť časového kroku [3].

3.2 Geodesic mean curvature flow (GMCF)

Tento model používa proces difúzie riadený krivosťou na filtrovanie dát. V tomto prístupe považujeme spracované dáta za súbor vrstevníc (izočiar). Vývoj týchto vrstevníc zodpovedá vyhladzovaniu spracovávaných dát.

3.2.1 Okrajová úloha s počiatočnou podmienkou

$$\frac{\partial u}{\partial t} = |\nabla u| \nabla \cdot \left(g \frac{\nabla u}{|\nabla u|} \right), \quad x \in \Omega, \quad t \in [0, \sigma], \quad (3.26)$$

$$\frac{\partial u}{\partial \vec{n}} = 0, \quad x \in \partial\Omega, \quad t \in [0, \sigma], \quad (3.27)$$

$$u(x, 0) = u^0(x), \quad x \in \Omega, \quad (3.28)$$

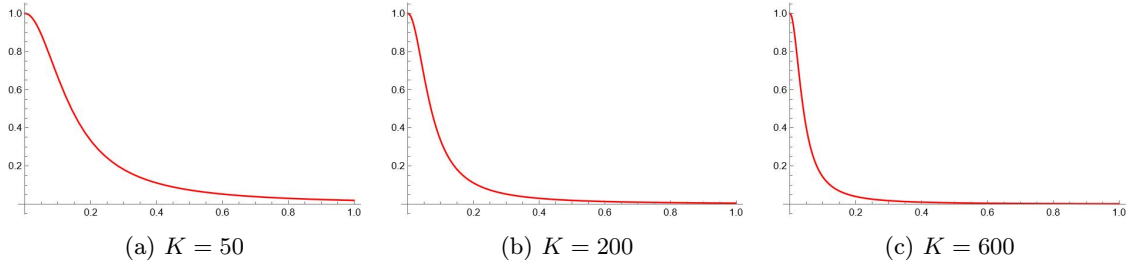
kde $u(x, t)$ je intenzita, \vec{n} je jednotková vonkajšia normála ku hranici $\partial\Omega$, funkcia g je difúzny koeficient a $u^0(x)$ je počiatočný obraz [3].

Difúzny koeficient g

Difúzny koeficient ovplyvňuje rýchlosť a intenzitu zhladzovania. V našom modeli je difúzny koeficient g daný funkciou hranového detektora v tvare

$$g(|\nabla u^\sigma|) = \frac{1}{1 + K|\nabla u^\sigma|^2}, \quad K \geq 0, \quad (3.29)$$

kde ∇u^σ je predvyhladený gradient riešenia rovnice vedenia tepla aplikovanej na u s časovým krokom σ . Hranový detektor vracia hodnoty v rozmedzí $[0, 1]$. Parameter K určuje, aká bude miera citlivosti detektora na vysoké hodnoty normy gradientu $|\nabla u^\sigma|$. Vizualizovali sme graf funkcie g pre rôzne hodnoty parametra K na obrázku Obr. 3.2. Čím vyššie K zvolíme, tým pomalšie sa budú zhladzovať oblasti (spomalí sa vyhladzovanie izočiar), kde sa nachádzajú veľké gradienty.



Obr. 3.2: Funkcia hranového detektora pre rôzne hodnoty parametra K pre dátový súbor s intenzitami v rozmedzí $[0, 1]$

Mean curvature flow

Ak funkcia $g = 1$, rovnica (3.26) má tvar

$$\frac{\partial u}{\partial t} = |\nabla u| \nabla \cdot \left(\frac{\nabla u}{|\nabla u|} \right), \quad x \in \Omega, \quad t \in [0, \sigma] \quad (3.30)$$

a predstavuje model Mean curvature flow (MCF), ktorý môžeme volať aj krivostný filter. Jeho cieľom je vyhladiť hrany a redukovať šum. Šum má veľkú krivosť, preto sa izočiare zodpovedajúce šumu stiahnu do bodu oveľa rýchlejšie ako izočiare reálnych objektov v obraze.

Numerické riešenie

Aby sme rovnicu (3.26) mohli numericky aproximovať, najskôr musíme regularizovať normu gradientu Evansovou-Spruckovou ϵ -regularizáciou

$$|\nabla u| \approx |\nabla u|^\epsilon = \sqrt{\epsilon^2 + |\nabla u|^2}, \quad (3.31)$$

kde $\epsilon \ll 1$ a zároveň $\epsilon > 0$. Tento krok je nutný kvôli tomu, aby sa v schéme nevyskytvalo delenie nulou.

Ďalej postupujeme rovnako ako pri rovnici vedenia tepla a budeme používať rovnaké značenie. Najskôr urobíme časovú diskretizáciu. Zvolíme časový krok τ a aproximujeme deriváciu podľa času spätnou diferenciou v tvare

$$\frac{\partial u}{\partial t} \approx \frac{u^n - u^{n-1}}{\tau}. \quad (3.32)$$

Difúzny koeficient g a normy gradientov z u počítame s riešením z predošlého časového kroku a gradient funkcie u budeme uvažovať z aktuálneho časového kroku. Z toho dostaneme

$$\frac{u^n - u^{n-1}}{\tau} = |\nabla u^{n-1}|^\epsilon \nabla \cdot \left(g^{n-1} \frac{\nabla u^n}{|\nabla u^{n-1}|^\epsilon} \right). \quad (3.33)$$

Následne rovnicu vynásobíme $\frac{1}{|\nabla u^{n-1}|^\epsilon}$ a dostaneme semi-implicitnú schému rovnice (3.26) v tvare

$$\frac{1}{|\nabla u^{n-1}|^\epsilon} \frac{u^n - u^{n-1}}{\tau} = \nabla \cdot \left(g^{n-1} \frac{\nabla u^n}{|\nabla u^{n-1}|^\epsilon} \right). \quad (3.34)$$

Rovnicu (3.34) zintegrujeme nad konečným objemom p a máme

$$\int_p \frac{1}{|\nabla u^{n-1}|^\epsilon} \frac{u^n - u^{n-1}}{\tau} dx = \int_p \nabla \cdot \left(g^{n-1} \frac{\nabla u^n}{|\nabla u^{n-1}|^\epsilon} \right) dx, \quad (3.35)$$

na pravej strane rovnice použijeme Gaussovú vetu a využijeme vlastnosť hranice ∂p a dostaneme

$$\int_p \frac{1}{|\nabla u^{n-1}|^\epsilon} \frac{u^n - u^{n-1}}{\tau} dx = \sum_{q \in N(p)} \int_{e_{pq}} \frac{1}{|\nabla u^{n-1}|^\epsilon} g^{n-1} \nabla u^n \cdot \vec{n}_p dx. \quad (3.36)$$

Aproximáciu objemového integrálu z regularizovného gradientu $|\nabla u^{n-1}|^\epsilon$ označíme $|\bar{\nabla} u^{n-1}|_p^\epsilon$, tok na hranici aproximujeme ako

$$\nabla u^n \cdot \vec{n}_{pq} = \frac{\partial u^n}{\partial \vec{n}_{pq}} \approx \frac{u_q^n - u_p^n}{d_{pq}} \quad (3.37)$$

a konštanty vyberieme pred integrály. Tým získame

$$\frac{m(p)}{|\bar{\nabla} u^{n-1}|_p^\epsilon} \frac{u^n - u^{n-1}}{\tau} = \sum_{q \in N(p)} m(e_{pq}) \frac{g^{n-1}}{|\nabla u^{n-1}|_{pq}^\epsilon} \frac{u_q^n - u_p^n}{d_{pq}} dx. \quad (3.38)$$

Pre pixelovú štruktúru platí $m(p) = m(e_{pq}) = 1$ a po úpravách dostaneme výslednú semi-implicitnú schému

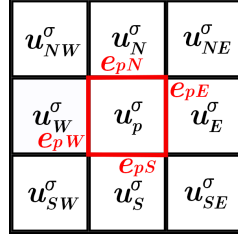
$$\left(1 + \tau |\bar{\nabla} u^{n-1}|_p^\epsilon \sum_{q \in N(p)} \frac{g_{pq}^{n-1}}{|\nabla u^{n-1}|_{pq}^\epsilon} \right) u_p^n - \tau |\bar{\nabla} u^{n-1}|_p^\epsilon \sum_{q \in N(p)} \frac{g_{pq}^{n-1}}{|\nabla u^{n-1}|_{pq}^\epsilon} u_q^n = u_p^{n-1}. \quad (3.39)$$

Nulovú Neumannovu podmienku zabezpečíme tým, že člen $u_q^n - u_p^n$ v rovnici (3.38) je rovný nule, ak p je bunka na hranici siete pixelov a q je bunka mimo siete pixelov.

Aproximáciu objemového integrálu z regularizovného gradientu $|\nabla u^{n-1}|^\epsilon$ vypočítame ako priemernú hodnotu z noriem gradientov na hranách konečného objemu

$$|\bar{\nabla} u^{n-1}|_p^\epsilon = \frac{\sum_{q \in N(p)} |\nabla u^{n-1}|_{pq}^\epsilon}{\sum_{q \in N(p)} 1}. \quad (3.40)$$

Ešte je potrebné definovať aproximáciu difúzneho koeficientu g . Najskôr musíme aproximovať vstupný parameter $|\nabla u^\sigma|$. Predhladené riešenie u^σ získame ako riešenie rovnice vedenia tepla s časovým krokom σ . Normu gradientu z u^σ aproximujeme pomocou konečných diferencií. Hrany pixela p označíme ako $e_{pE}, e_{pW}, e_{pN}, e_{pS}$ a hodnoty u^σ v okolí u_p^σ označíme ako $u_{NE}^\sigma, u_{NW}^\sigma, u_{SE}^\sigma, u_{SW}^\sigma, u_N^\sigma, u_S^\sigma, u_E^\sigma, u_W^\sigma$ (Obr. 3.3).



Obr. 3.3: Označenie hrán pixela p a jeho susedov

Urobíme výpočet na hrane e_{pE}

$$\left. \frac{\partial u^\sigma}{\partial x} \right|_{e_{pq}=e_{pE}} \approx \frac{u_E^\sigma - u_p^\sigma}{h}, \quad (3.41)$$

$$\left. \frac{\partial u^\sigma}{\partial y} \right|_{e_{pq}=e_{pE}} \approx \frac{1}{2} \left(\frac{u_N^\sigma - u_S^\sigma}{2h} + \frac{u_{NE}^\sigma - u_{SE}^\sigma}{2h} \right), \quad (3.42)$$

$$|\nabla u^\sigma|_{pE}^2 \approx \left(\frac{u_E^\sigma - u_p^\sigma}{h} \right)^2 + \left(\frac{u_N^\sigma - u_S^\sigma + u_{NE}^\sigma - u_{SE}^\sigma}{4h} \right)^2 \quad (3.43)$$

Analogicky budeme postupovať aj pri zvyšných hranách. Rovnakým spôsobom vieme aproximovať aj regularizované gradienty $|\nabla u^{n-1}|_{pq}^\epsilon$, avšak s tým rozdielom, že nepočítame s predhladeným riešením u^σ , ale s pôvodným riešením u a výsledok $|\nabla u|^2$ následne aproximujeme pomocou vzorca (3.31) [3].

Kapitola 4

Izočiary

Po tom, čo sme použili filtráciu, sú na snímke viditeľné hlavné štruktúry oblastí svetelného znečistenia. Môžeme prejsť na ďalší krok, a to segmentáciu regiónov, ktoré tvoria mestá. Následne nájdeme ich hranice, ktoré budú reprezentované polygónmi. Na to použijeme upravený Marching squares algoritmus [4], ktorý nájde hranicu mesta ako orientovanú izočiaru, ktorá spája hodnoty intenzity žiarenia na hranici mesta. Ak ide o vonkajšiu hranicu mesta je orientovaná proti smeru hodinových ručičiek, zatiaľ čo pri vnútornej hranici je orientovaná v smere hodinových ručičiek. Tento algoritmus je vhodný pre uzavreté izočiary, ktoré sú vzdialené od hranice obrázku aspoň 2 pixely.

4.1 Algoritmus na detekciu izočiari

Algoritmus sa skladá z nasledujúcich krokov:

1. Zvolíme prahovú hodnotu iso a z pôvodnej snímky s intenzitami $u(i, j)$ vytvoríme binárny obrázok s hodnotami $b(i, j)$ nasledujúcim spôsobom

$$b(i, j) = \begin{cases} 0, & \text{ak } u(i, j) < iso \\ 1, & \text{ak } u(i, j) \geq iso. \end{cases} \quad (4.1)$$

2. Následne prechádzame binárnym obrázkom, kým neprídeme nakoniec a hľadáme referenčný pixel (i, j) taký, že $b(i, j) = 0$ a $b(i + 1, j + 1) = 1$ pričom musí platiť, že pixel (i, j) alebo $(i + 1, j + 1)$ ešte nebol prehľadávaný pri hľadaní niektorej inej izočiary.

Po jeho nájdení vykonáme nasledujúce kroky:

- (a) Od referenčného pixelu (i, j) prehľadávame bloky pixelov 2×2 , pričom prehľadávaný pixel predstavuje ľavý horný roh bloku. Značíme si už prehľadané pixely blokov. Súradnice prehľadávaného pixelu budeme značiť (x, y) .
- (b) Bloku pixelov priradíme číslo (rozhodovacie kritérium) podľa toho, či sú jednotlivé pixely v bloku čierne alebo biele.

```
sum = 0;
if (b[y][x]) sum += 1;
if (b[y][x + 1]) sum += 2;
if (b[y + 1][x]) sum += 4;
if (b[y + 1][x + 1]) sum += 8;
```

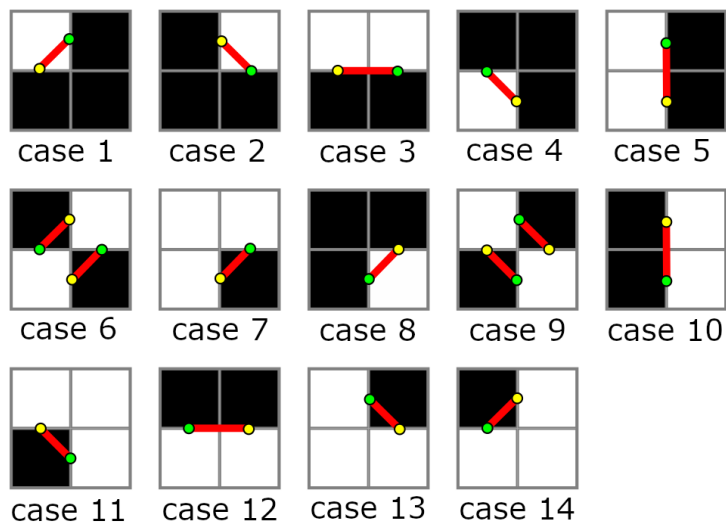
- (c) Podľa rozhodovacieho kritéria sa rozhodneme o smere posunu referenčného pixela.


```

switch (sum) {
case 1:direction = N; break;
case 2:direction = E; break;
case 3:direction = E; break;
case 4:direction = W; break;
case 5:direction = N; break;
case 6:direction = previous == N ? E : W; break;
case 7:direction = E; break;
case 8:direction = S; break;
case 9:direction = previous == E ? S : N; break;
case 10:direction = S; break;
case 11:direction = S; break;
case 12:direction = W; break;
case 13:direction = N; break;
case 14:direction = W; break;
default: qDebug() << "problem\n"; exit;
}

```

- (d) Na obrázku Obr. 4.1 sú zobrazené jednotlivé prípady, ktoré môžu nastať na základe rozhodovacieho kritéria z kroku (b). Je tu zobrazený blok pixelov 2×2 , červenou farbou je znázornená izočiara a žltou je znázornený bod izočiary, ktorý sme našli pre predošlý prehľadávaný pixel. Pokiaľ je prehľadávaný pixel zároveň referenčným pixelom, teda prvým prehľadávaným pixelom, uložíme oba body izočiary (zelený aj žltý). Inak uložíme súradnice bodu izočiary, ktorý znázornený zelenou farbou.



Obr. 4.1: Súradnice izočiary pre jednotlivé prípady

- (e) Ak sa prehľadávaný pixel nachádza na hranici obrázka, preskočíme nasledujúce dva kroky.
- (f) Posunieme prehľadávaný pixel na základe smeru z kroku (c).

```

switch (direction) {
case N: y -= 1; break;
case S: y += 1; break;
case E: x += 1; break;

```

```
case W: x -= 1; break;
}
```

- (g) Ak je prehľadávaný pixel (x, y) totožný s referenčným pixelom (i, j) pokračujeme v kroku 2, inak sa vrátíme ku kroku (a).

3. Po prejdení celého obrázka skončíme.

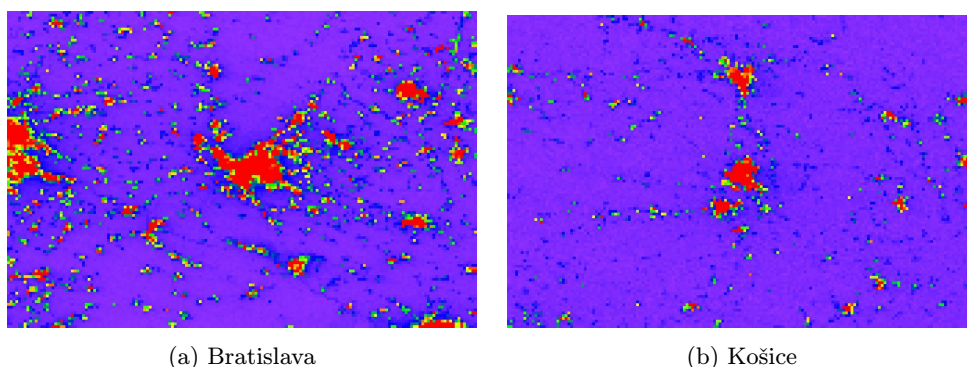
Aby sme tento algoritmus mohli použiť, musíme zvoliť takú hodnotu *iso*, aby zodpovedala intenzite žiarenia na hranici mesta. Voľba tejto hodnoty je ponechaná na užívateľovi aplikácie. Keďže hľadáme izočiary, ktoré reprezentujú mestá, je potrebné, aby bola izočiara uzavretá, preto neuvažujeme izočiary, ktoré končia na hranici obrázka alebo sa jej dotýkajú. Súradnice izočiary sa nachádzajú vždy v strede strany pixelu, preto ich ukladáme ako neceločíselné hodnoty.

Kapitola 5

Numerické experimenty

Na riešenie lineárnej sústavy rovníc (3.25) sme použili metódu Successive Over-Relaxation (SOR) [10], upravenú pre našu úlohu, pričom sme zvolili superrelaxačný koeficient $\omega = 1.25$. Na riešenie sústavy rovníc (3.39) sme použili metódu Biconjugate gradient stabilized (BiCGSTAB), ktorú sme implementovali pomocou knižnice Eigen [1].

Pristúpime k filtrácii dvoch testovacích snímok Obr. 5.1a a Obr. 5.1b, pričom v strede prvej snímky sa nachádza mesto Bratislava a v strede druhej mesto Košice. Na zobrazenie snímok sme použili farebnú paletu Rainbow s rozsahom intenzít $[0, 100]$ a obrázky sme preškálovali pre lepšiu viditeľnosť na 5-násobnú veľkosť. Teda namiesto jedného pixela s farbou f vykreslíme blok pixelov 5×5 tou istou farbou f .



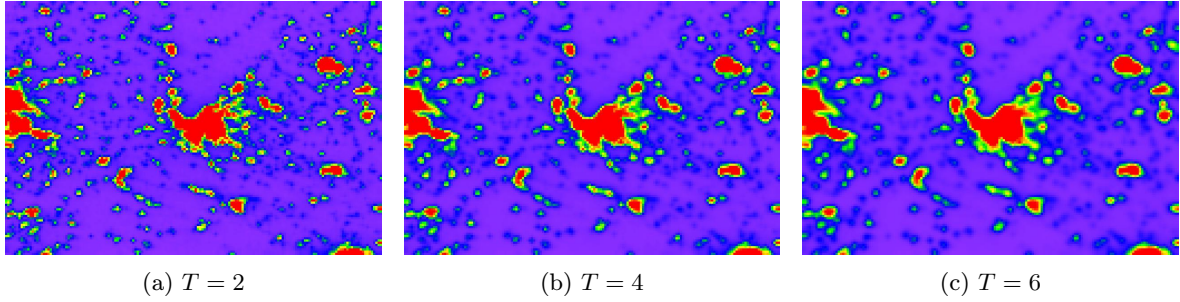
Obr. 5.1: Vizualizácia satelitných snímok intenzity svetelného žiarenia

5.1 Filtrácia lineárnou difúziou

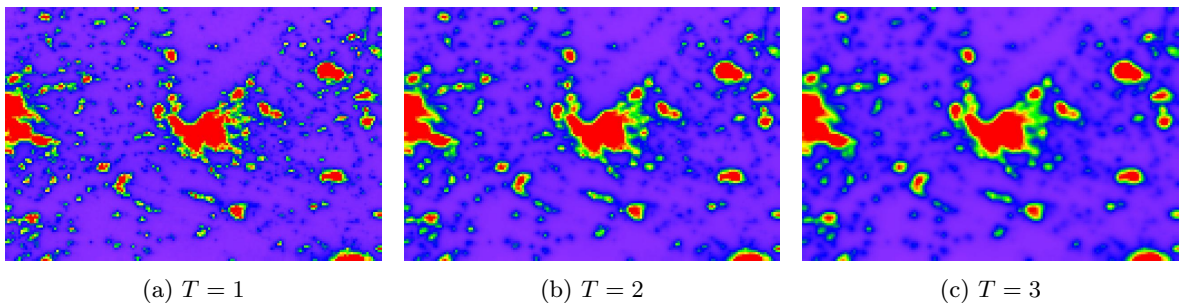
Začneme s jednoduchšou metódou filtrácie, a to lineárnou filtráciou. Na výpočet riešenia sme použili explicitnú schému (3.16) a implicitnú schému (3.25). Výber schémy závisí na časovom kroku τ , pričom ak $\tau > 0.2$, použili sme implicitnú schému a v opačnom prípade sme použili explicitnú schému.

Snímku 5.1a sme filtrovali s počtom časových krokov $T = 6$, časovým krokom $\tau = 0.1$ a na výpočet sme použili explicitnú schému. Riešenie je zobrazené po časových krokoch $T = 2, 4, 6$ (Obr. 5.2). Na filtráciu s parametrami $T = 3$ a $\tau = 0.4$ sme použili explicitnú schému a na obrázku Obr. 5.3 sme zobrazili riešenie po časových krokoch $T = 1, 2, 3$.

Zo snímok sme odstránili šum, avšak veľmi malé oblasti svetelného znečistenia sa nám nepodarilo odstrániť, hrany prestali byť ostré a zväčšila sa plocha mesta oproti pôvodnej snímke. Preto pristúpime k zložitejšej, ale pre našu úlohu vhodnejšej, metóde nelineárnej filtrácie.



Obr. 5.2: Filtrácia lineárnou difúziou pre časové kroky $T = 2, 4, 6$ s časovým krokom $\tau = 0.1$



Obr. 5.3: Filtrácia lineárnou difúziou pre časové kroky $T = 1, 2, 3$ s časovým krokom $\tau = 0.4$

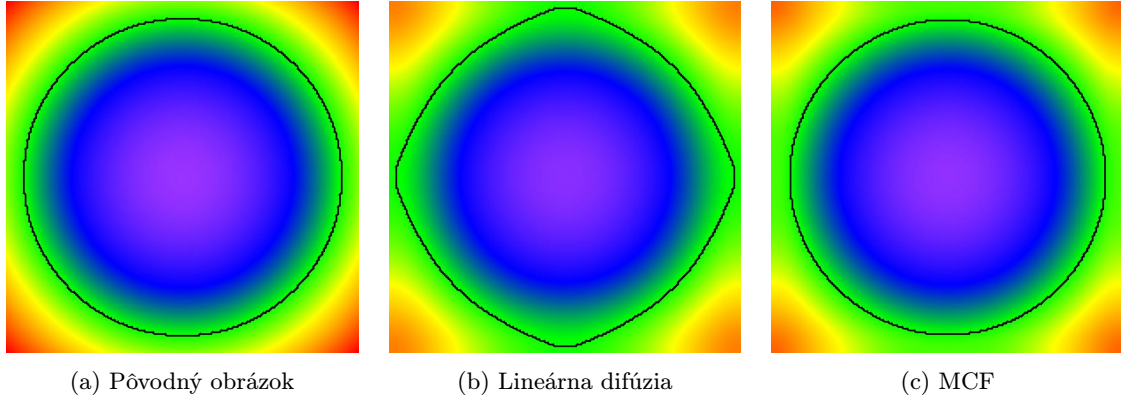
5.2 Filtrácia pomocou GMCF

Vplyv krivostného filtra znázorníme pomocou modelu MCF, ktorý je zjednodušením modelu GMCF, kde funkcia $g = 1$, čo vieme dosiahnuť parametrom $K = 0$. Na výpočet MCF použijeme semi-implicitnú schému GMCF (3.39), kde $g_{pq}^{n-1} = 1$. V tejto schéme počítame s regularizovanými normami gradientov $|\nabla u|^\epsilon$, ktoré závisia od parametra ϵ . Pre $\epsilon \ll 1$ sa zhladzujú izočiarly a pre väčšie ϵ sa zhladzuje celá funkcia intenzity, teda filtrácia má podobný účinok ako lineárna difúzia. My použijeme parameter $\epsilon = 0.001$, ktorý vizuálne zachováva vlastnosti MCF a zároveň lineárna sústava nebude príliš náročná na výpočet.

Ako testovací obrázok sme použili obrázok Obr. 5.4a, ktorý predstavuje graf funkcie

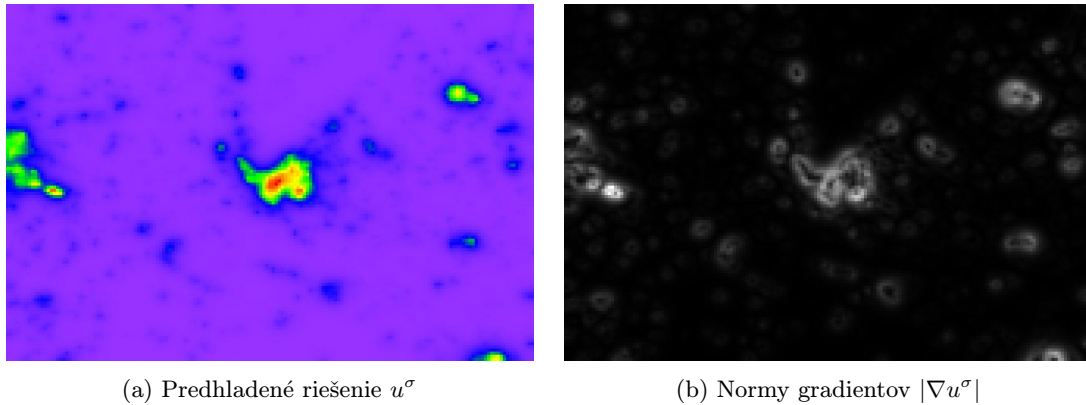
$$(x, y) = x^2 + y^2, \quad x \in [-10, 10], \quad y \in [-10, 10],$$

kde je čiernou farbou znázornená izočiarla s hodnotou 82. Našli sme ju pomocou Algoritmu na detekciu izočiar zo sekcie 4.1 a ďalšiemu jeho použitiu sa venujeme v sekcii 5.3. Porovnali sme jeho filtráciu pomocou lineárnej difúzie (Obr. 5.4b) s filtráciou pomocou MCF (Obr. 5.4c) pre parametre $T = 10$ a $\tau = 10$.

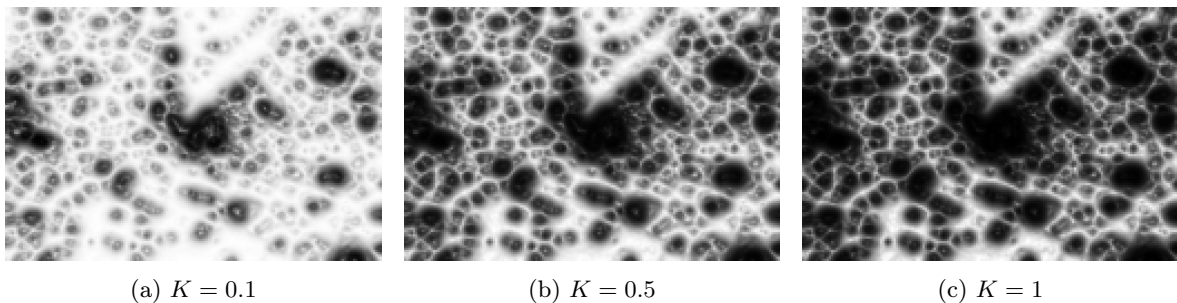


Obr. 5.4: Porovnanie filtrácie lineárnou difúziou a MCF

Pôvodná kružnicová izočiara použitím lineárnej difúzie stratila svoj tvar, zatiaľ čo pri MCF si ho zachovala. Model MCF zhladzuje rovnomerne podľa strednej krivosti, čiže sťahuje izočiary do kružnice. Ak chceme, aby zhladzovanie bolo adaptívne, použijeme hranový detektor g , čo má za následok pomalšie zhladzovanie oblastí s veľkými gradientmi. Vstupný parameter do hranového detektora g je norma gradientu z predhladeného riešenia u^σ . Pre snímku Obr. 5.1 sme zobrazili predhladené riešenie u^σ , kde $\sigma = 2.5$ (Obr. 5.6a). Normy gradientov z predhladeného riešenia $|\nabla u^\sigma|$ (Obr. 5.12b) a hodnoty hranového detektora g pre rôzne hodnoty parametra K (Obr. 5.6) sme zobrazili vo farebenej palete Greyscale, kde biela farba predstavuje najvyššiu hodnotu a čierna najnižšiu.



Obr. 5.5: Vizualizácia predhladeného riešenia a noriem gradientov z predhladeného riešenia

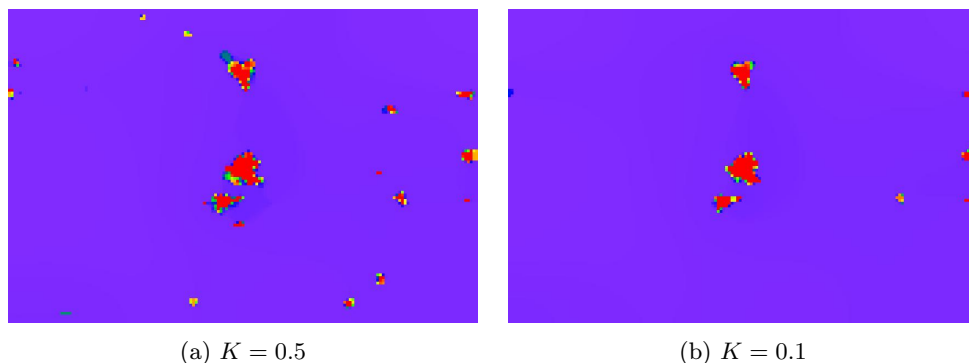


Obr. 5.6: Mapa hodnôt hranového detektora g pre rôzne hodnoty parametra K

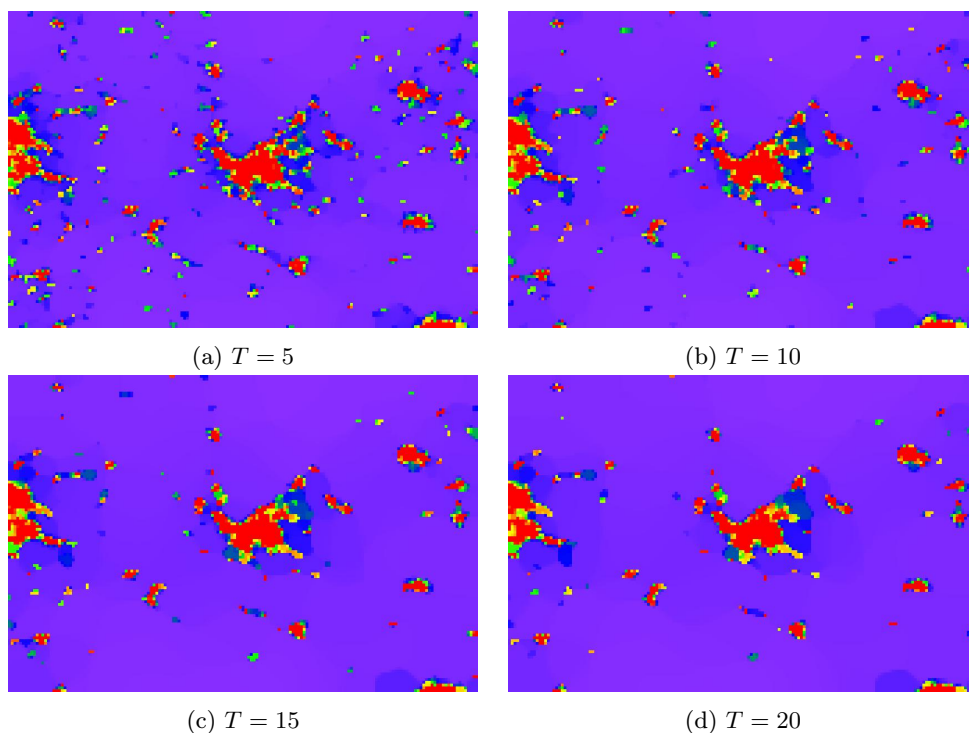
Na filtráciu pomocou GMCF sme použili semi-implicitnú schému (3.39). Na obrázku Obr. 5.8 je zobrazené riešenie filtrácie snímky 5.1a s parametrami $K = 0.5$, $\epsilon = 0.001$, $\sigma = 0.5$, $\tau = 10$ a $T = 20$ pre časové kroky $T = 5, 10, 15, 20$. Na obrázku Obr. 5.9 je zobrazená filtrácia

toho istého snímku s parametrom $K = 0.1$, pričom zvyšné parametre ostali rovnaké. Na obrázku 5.7 sme zobrazili filtráciu snímky 5.1b v časovom kroku $T = 20$ a parametrami $\epsilon = 0.001$, $\sigma = 0.5$, $\tau = 10$, pričom na obrázku 5.7a sme pri filtrácii použili parameter $K = 0.5$ a na obrázku 5.7b parameter $K = 0.1$.

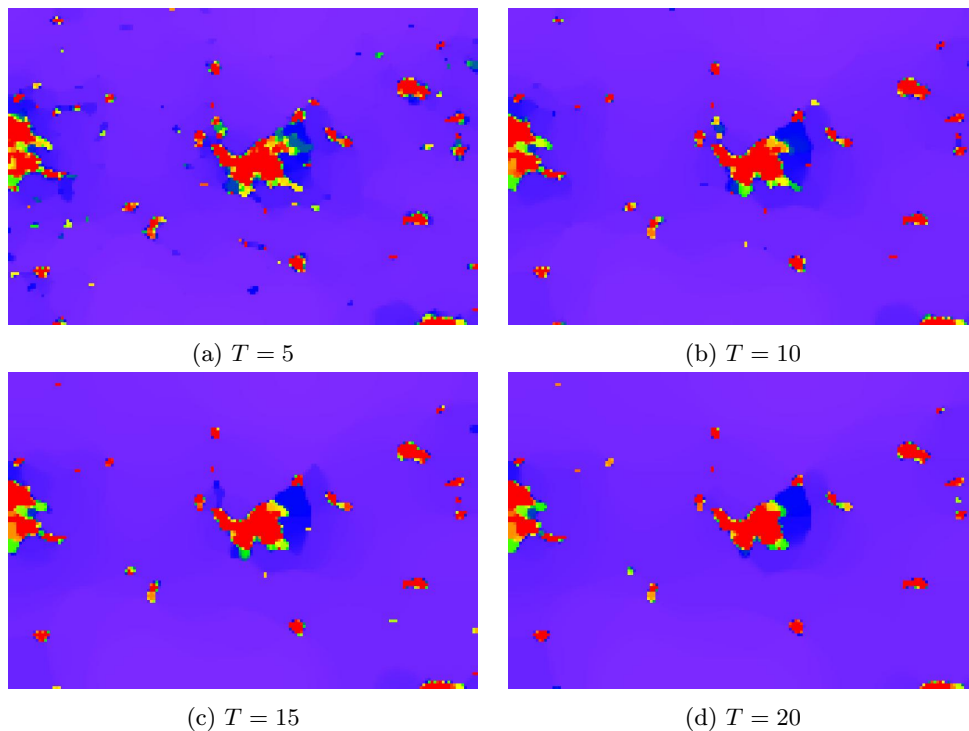
Po filtrácii s parametrom $K = 0.5$ sa zachovali aj rozlohou menšie oblasti a oblasti s nižšou intenzitou žiarenia. Mestá si zachovali vnútornú štruktúru a môžeme pozorovať oblasti z nižšou intenzitou na hranici mesta. Zatiaľ čo po filtrácii s parametrom $K = 0.1$ pozorujeme rýchlejšie vyhladenie šumu a ostrejšie hrany miest, pričom sa zhladzuje aj vnútorná štruktúra.



Obr. 5.7: Filtrácia snímky Obr. 5.1b pomocou GMCF pre počet časových krokov $T = 20$ s rôznymi parametrami K



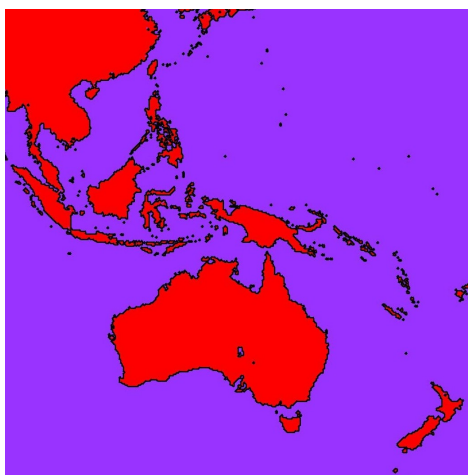
Obr. 5.8: Filtrácia snímky Obr. 5.1a pomocou GMCF pre rôzne počty časových krokov T s parametrom $K = 0.5$



Obr. 5.9: Filtrácia snímky Obr. 5.1a pomocou GMCF pre rôzne počty časových krokov T s parametrom $K = 0.1$

5.3 Izočiary

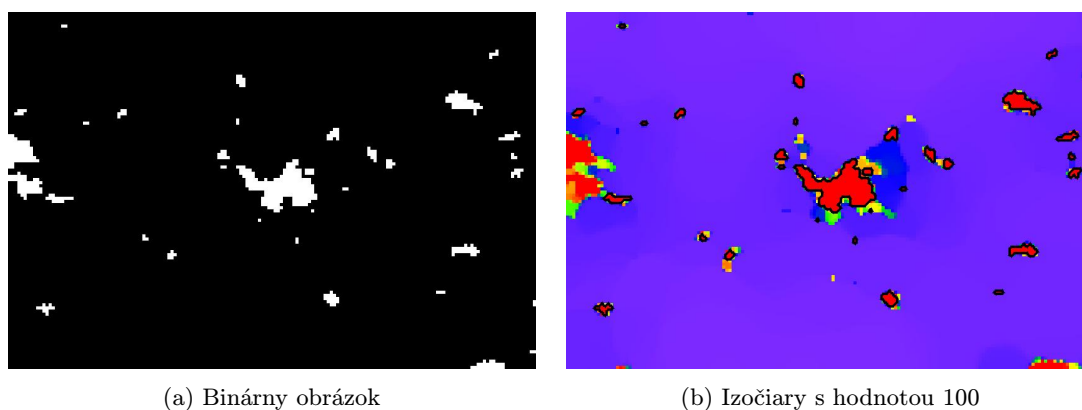
Aby sme našli hranice miest použili sme upravený Marching squares algoritmus z kapitoly 4. Tento algoritmus sme najskôr testovali na mape zemského povrchu, kde hodnota 1 predstavuje pevninu a hodnota 0 predstavuje vodné plochy. Na obrázku Obr. 5.10 sme zobrazili výsek z tejto mapy, kde červená farba predstavuje pevninu, fialová vodné plochy a čiernou farbou je zobrazená hranica pevniny, ktorú sme našli ako izočiary s prahovou hodnotou $iso = 1$. Aby sme mohli vykresliť súradnice izočiary, ktoré sú neceločíselné, preškálujeme veľkosť obrázka parametrom s , kde $s \in \mathbb{N}$ a $s \geq 1$. Pixely súradníc izočiary nájdeme ich vynásobením parametrom s a následným celočíselným zaokrúhlením.



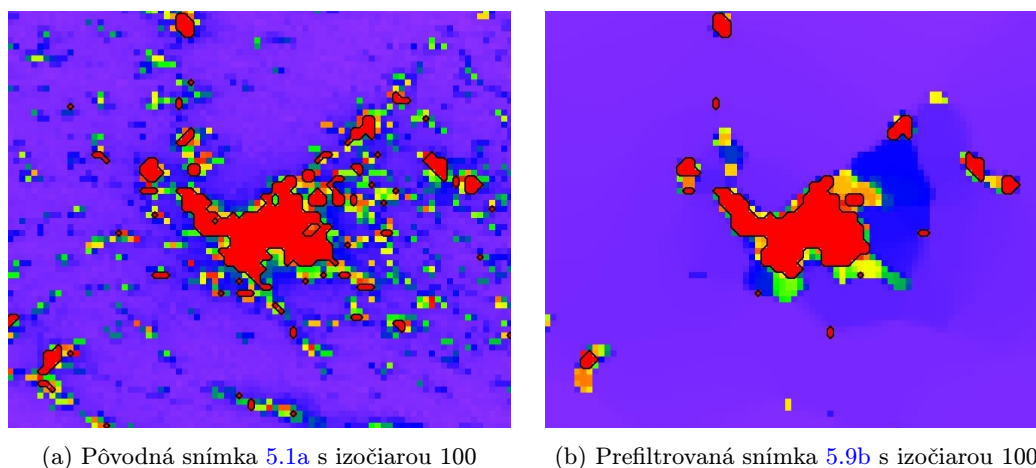
Obr. 5.10: Výseky z mapy zemského povrchu s izočiarami

Ako vstup do algoritmu sme použili prefiltrovanú snímku mesta Bratislavy a jej okolia

(Obr. 5.9b) s prahovacou hodnotou $iso = 100$. Algoritmus vysegmentuje zo snímky mestá prahovacou metódou, pričom vytvorí binárny obrázok 5.11a. Izočiara s hodnotou 100 je hranicou objektov znázornených bielou farbou. Následne sme vykreslili izočiary uzavretých objektov spolu s prefiltrovanou snímku na obrázku Obr. 5.11b. Porovnali sme izočiary neprefiltrovaného snímku s prefiltrovaným (Obr. 5.12). Filtráciou sme odstránili vnútorné izočiary mesta, teda oblasti ohraničené polygónmi sú súvislé. Hranice miest sa vyhladili a veľmi malé oblasti sa odstránili.

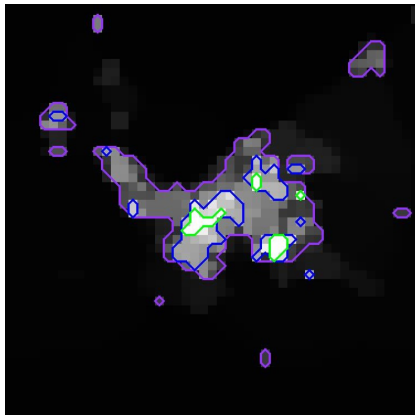


Obr. 5.11: Algoritmus na detekciu izočiari

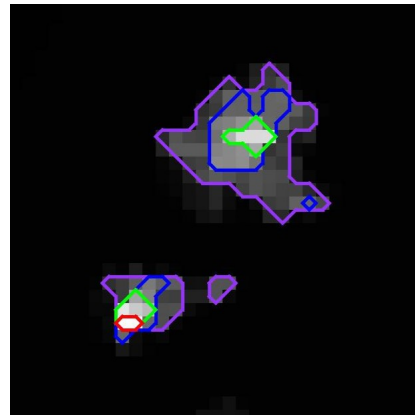


Obr. 5.12: Výrezy zo snímok s izočiariami

Na obrázku Obr. 5.13 sme zobrazili viaceré izočiary v jednom meste. Na obrázku Obr. 5.13a je výrez z prefiltrovanej snímky Bratislavy a okolia (Obr. 5.1a) pomocou GMCF s parametrami $K = 0.1$, $\epsilon = 0.001$, $\sigma = 0.5$, $\tau = 10$ a $T = 10$, zobrazená v paleta Grayscale z rozsahom intenzít $[0, 600]$, kde fialová izočiara predstavuje hodnotu 100, modrá 350 a zelená 600. Na obrázku Obr. 5.13b je výrez snímky Košíc a okolia (Obr. 5.1b) prefiltrovaný tou istou metódou a parametrami ako obrázok Obr. 5.13a, ale zobrazený v paleta Grayscale s rozsahom intenzít $[0, 650]$, pričom fialová izočiara predstavuje hodnotu 100, modrá 250, zelená 400 a červená 650.



(a) Bratislava



(b) Košice

Obr. 5.13: Viaceré izočiary v jednom meste

Kapitola 6

Záver

V tejto práci sme sa venovali spracovaniu dát získaných zo satelitného senzoru VIIRS, difúznej a krivostnej filtrácii a hľadaniu izočiar za účelom vytvoriť polygonálne modely miest. V kapitole 2 sme previedli satelitné snímky do formátu ASCII grid, v ktorom sú dáta usporiadané v pravidelnej mriežke. V sekcii 2.2 sme formulovali algoritmus na vykreslenie mapy intenzity s ľubovoľnou farebnou škálou a ľubovoľným rozsahom intenzít. V kapitole 3 sme formulovali numerické schémy na výpočet difúznej a krivostnej filtrácie, ktoré sme riešili explicitne alebo numerickými iteračnými metódami. Segmentáciu miest a následnú detekciu ich hraníc vo forme polygónov sme definovali v kapitole 4. Funkčnosť algoritmov a numerických schém sme demonštrovali v kapitole 5. Filtráciou sa nám podarilo získať homogénne oblasti, ktoré sme segmentovali a našli ich hranice. Výsledok tohto postupu závisí od správne zvolených parametroch matematických modelov. Ich voľba by mala zodpovedať zámerom užívateľa aplikácie.

Priestor na zlepšenie sa nachádza v implementovanej pokročilejšej prahovacej metódy, vhodnej pre obrazy s nejednotným osvetlením. Jedna z takých metód je Niblackova metóda, ktorá môže vrátiť viacero prahových hodnôt, pretože každý pixel v obraze je segmentovaný pomocou prahovej hodnoty vypočítanej z jeho lokálneho okolia. Dokazáli by sme tak vysegmentovať mestá s rôznymi hodnotami prahu.

Ďalším krokom bude exportovanie polygónov do formátu XML, ktorý slúži ako vstup do nástroja SkyGlow simulator, výpočet štatistických charakteristík polygónov a porovnanie modelov miest s modelmi získanými ručnou segmentáciou výpočtom Hausdorffovej vzdialenosti.

Bibliografia

1. *Eigen*. Dostupné tiež z: https://eigen.tuxfamily.org/index.php?title=Main_Page.
2. KOCIFAJ, M. et al. *SkyGlow - SAV*. Dostupné tiež z: <http://www.skyglow.sav.sk/>.
3. KRIVÁ, Z.; MIKULA, K.; STAŠOVÁ, O. *Spracovanie obrazu: Vybrané kapitoly z prednášok*. Slovenská technická univerzita v Bratislave, 2016.
4. KÁČER, K. *Detekcia orientovaných izočiar v segmentovanom obraze a ich úprava na základe štatistiky*. Bratislava, 2023. Bakalárska práca. Slovenská technická univerzita.
5. NASA. *Suomi National Polar-orbiting Partnership (Suomi NPP)*. Dostupné tiež z: <https://eosps.nasa.gov/missions/suomi-national-polar-orbiting-partnership>.
6. NASA. *Visible Infrared Imaging Radiometer Suite (VIIRS)*. Dostupné tiež z: <https://www.earthdata.nasa.gov/sensors/viirs>.
7. NASA. *VNP46A1 - VIIRS/NPP Daily Gridded Day Night Band 500m Linear Lat Lon Grid Night*. Dostupné tiež z: <https://ladsweb.modaps.eosdis.nasa.gov/missions-and-measurements/products/VNP46A1/#overview>.
8. NOAA. *VIIRS EDR Imagery*. Dostupné tiež z: https://www.star.nesdis.noaa.gov/portfolio/detail_VIIRS_GTM_NCC.php.
9. SEAMAN, C. *Beginner's Guide to VIIRS Imagery Data*. 2013. Dostupné tiež z: https://rammb.cira.colostate.edu/projects/npp/Beginner_Guide_to_VIIRS_Imagery_Data.pdf.
10. *Successive over-relaxation method - SOR*. Dostupné tiež z: https://www.cfd-online.com/Wiki/Successive_over-relaxation_method_-_SOR.
11. *Surfer*. Dostupné tiež z: <https://www.goldensoftware.com/products/surfer/>.
12. THE HDF GROUP. *THE HDF5® LIBRARY FILE FORMAT*. Dostupné tiež z: <https://www.hdfgroup.org/solutions/hdf5/>.
13. THE QT COMPANY. *Qt — Tools for Each Stage of Software Development Lifecycle*. Dostupné tiež z: <https://www.qt.io/>.