

SLOVENSKÁ TECHNICKÁ UNIVERZITA V BRATISLAVE
STAVEBNÁ FAKULTA

Automatické segmentačné metódy biologických dát

Diplomová práca

Študijný program:	Matematicko-počítačové modelovanie
Študijný odbor:	Aplikovaná matematika
Školiace pracovisko:	Katedra matematiky a deskriptívnej geometrie
Vedúci diplomovej práce:	doc. RNDr. Zuzana Krivá, PhD.

BRATISLAVA 2020
Bc. Mária Somorovská



ZADANIE DIPLOMOVEJ PRÁCE

Študentka: **Bc. Mária Somorovská**
ID študenta: 82218
Študijný program: matematicko-počítačové modelovanie
Študijný odbor: matematika
Vedúca práce: doc. RNDr. Zuzana Krivá, PhD.
Miesto vypracovania: Svf STU Bratislava

Názov práce: **Automatické segmentačné metódy biologických dát**

Jazyk, v ktorom sa práca vypracuje: slovenský jazyk

Špecifikácia zadania:

Cieľom práce je navrhnuť softvérové užívateľské prostredie pre skúmanie rôznych segmentačných metód pre biologické dátá a ich využitie v segmentačnom modeli SUBSURF, založenom na parciálnej diferenciálnej rovnici. Budú sa skúmať zavedené automatické postupy pomocou rôznych prahovacích metód a ich využitie v segmentačnom modeli SUBSURF, založenom na parciálnej diferenciálnej rovnici. Na implementáciu a vytvorenie prostredia bude zvolený objektovo orientovaný model jazyka C++, spolu s knižnicami Qt, ktoré obsahujú veľa užitočných tried na vytváranie užívateľského prostredia. Budú použité aj VTK knižnice, ktoré slúžia na zobrazovanie dát a manipuláciu s nimi.

Rozsah práce: 30-40 strán

Riešenie zadania práce od: 30. 10. 2019
Dátum odovzdania práce: 14. 05. 2020

Bc. Mária Somorovská
študentka

prof. RNDr. Radko Mesiar, DrSc.
vedúci pracoviska

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

Čestné vyhlásenie

Čestne vyhlasujem, že diplomou prácu som vypracovala samostatne, na základe konzultácií so školiteľkou a s použitím uvedených informačných zdrojov a literatúry.

V Bratislave, ?.?. 2020

.....
podpis autora práce

Abstrakt

Práca je zameraná na spracovanie biologických dát - makrofágov, ktoré môžu mať zložité tvary, šum a časti makrofágov môžu mať slabšiu intenzitu. Popisuje viacero globálnych a lokálnych adaptívnych prahovacích metód a ich využitie v segmentačnej metóde SUBSURF, založenej na parciálnej diferenciálnej rovnici. Práca je rozdelená do niekoľkých častí, teoretickú, ktorá popisuje využité segmentačné algoritmy z matematického hľadiska, praktickú, popisujúcu softvér slúžiaci na zobrazenie a spracovanie biologických dát. Na implementáciu segmentačných metód bol zvolený objektovo-orientovaný prístup v jazyku C++, spolu s Qt knižnicami slúžiacimi na vytvorenie užívateľského prostredia a s VTK knižnice boli využité pri vizualizácii. Práca obsahuje aj demonštráciu získaných výsledkov.

Kľúčové slová: spracovanie obrazov, C++, segmentačné metódy, prahovacie metódy.

Abstract

This thesis is dedicated to processing biological data of macrophages, which have difficult shapes, contain noise or on some parts can have lower intensity. The thesis describes numerous global and local adaptive thresholding methods and their usage in segmentation method SUBSURF, based on partial differential equation. The thesis is devided into several parts, where the theoretical one describes used segmentation algorithms from mathematical point of view and the practical one contains description of software used for visualization and processing the biological data. For the software implementation object oriented language C++ were used. User graphical interface was implemented with Qt libraries and VTK libraries were used for visualization. The work also contains results, which were achieved.

Key words: image processing, C++, segmentation methods, thresholding methods.

Obsah

1	Úvod	1
2	Segmentačná metóda	2
2.1	Globálne prahovanie	2
2.1.1	Otsuho metóda	2
2.1.2	Prahovanie pomocou maximálnej entropie	4
2.2	Lokálne adaptívne prahovanie	5
2.2.1	Bernsenova metóda	5
2.2.2	Niblackova metóda	6
2.2.3	Hybridné metódy	7
2.3	Metóda subjektívnych plôch (SUBSURF)	7
2.3.1	Modifikácia metódy	8
3	Numerické schémy	9
3.1	Implicitná metóda pre rovnicu vedenia tepla	9
3.2	Numerická schéma pre SUBSURF	10
4	Softvér	12
4.1	Qt	12
4.2	VTK	13
4.3	Rozdelenie projektu do tried	14
4.4	Grafické užívateľské rozhranie	15
4.5	Príklad použitia	18
5	Výsledky	22
6	Záver	34

1 Úvod

V rôznych vedných disciplínach ale aj v bežnom živote sa používajú rôzne aplikácie spracovania obrazov. Spracovanie obrazov je metóda, ktorá pomocou rôznych matematických operácií a algoritmov upravuje obrazové dátá rôznych formátov a pomáha z nich získať užitočné informácie. Obrazové dátá je potrebné zobraziť, či už pred alebo po modifikácii.

Cieľom predpokladanej práce je vytvorenie softvéru, ktorý slúži na vizualizáciu a segmentáciu obrazov získaných konfokálnym laserovým mikroskopom, konkrétnie sa jedná o biologické dátá a to obrazy makrofágov. Takéto dátá môžu obsahovať šum, ktorý je potrebné odstrániť pre lepšie rozoznanie objektov na dátach.

Z biologického hľadiska, je makrofág typ bielej krvinky, ktorá hrá dôležitú úlohu pri ochrane imunitného systému a hemostázy. Avšak disfunkčné makrofágy menia svoje účinky a u ľudí môžu spôsobovať závažné ochorenia ako sú napríklad zápalové ochorenia, ktoré vedú k častým infekciám alebo sa môžu podieľať na postupe rakoviny. Makrofág mení svoj tvar keď sa približuje smerom k rane. Táto zmena tvaru je spôsobená objektami, ktoré sa nachádzajú v blízkosti makrofágu, ako napríklad tkanivovými bunkami alebo medzibunkovou hmotou. Segmentácia obrazov môže byť užitočným nástrojom na porozumenie spôsobu interakcie medzi makrofágmi a bunkami, ktoré ho obklopujú, avšak takáto segmentácia môže byť náročnou úlohou, kvôli ich nepravidelnému tvaru a meniaci sa intenzite získaných obrazových dát. V tejto práci sa zaoberáme časťami/výrezmi takýchto dát, na ktorých sa nachádza jeden makrofág. V závere sme uviedli príklad segmentácie dát s viacerými makrofágmi.

Naimplementovaný model vychádza z práce [3], kde je segmentačná metóda založená na kombinácii prahovacích metód a segmentačnej metóde subjektívnych plôch(SUBSURF). Obe prahovacie metódy predstavovali globálne prahovanie - celý obrázok sa prahoval jednou hodnotou. V práci sme skúmali aj použitie lokálnych adaptívnych prahovacích metód, pri ktorých sa vytvára mapa prahov, pričom úseky so slabšou intenzitou sa prahujú s inými hodnotami ako úseky so silnejšou intenzitou. Predpokladali sme, že takéto metódy umožňujú lepšiu modifikáciu vektorového poľa advekčnej časti rovnice a zabezpečia lepšiu súvislosť výsledných tvarov, prípadne sa dajú použiť aj na tvorbu počiatočnej segmentačnej funkcie.

Našou snahou bolo vytvoriť softvér, ktorý by mal byť intuitívny, a teda určený aj užívateľom, ktorí implementovaným algoritmom nemusia rozumieť.

Práca je rozdelená do viacerých častí, v ktorých je podrobnejšie popísaná funkcia programu spolu s užívateľským prostredím, použitými algoritmami, knižnicami.

Prvá časť je teoretická a venuje sa matematickým algoritmom využitým pri implementácii. Jedná sa o niekoľko automatických a semi-automatických prahovacích metód, ktoré budú súčasťou segmentačnej metódy subjektívnych plôch(SUBSURF), pre ktorú je uvedený model, jej modifikácia a semi-implicitná konečno-objemová schéma.

Druhá časť sa zameriava na technológie a knižnice využité pri implementácii. Nachádza sa tam popis Qt knižníc, ktoré boli použité pri vytváraní užívateľského prostredia, VTK knižníc, ktoré boli využité na zobrazenie a manipuláciu s dátami. Popísané sú tu aj triedy, ktoré boli v programe najviac využité.

Ďalšia časť sa zaoberá popisom grafického rozhrania programu, ktorá by mohla slúžiť aj ako manuál slúžiaci užívateľovi pri používaní softvéru.

V poslednej časti sa nachádzajú výsledky, ku ktorým sme v práci dospeli a porovnania medzi rôznymi metódami prahovania a tvorby počiatočnej podmienky použitými v segmentačnom modeli.

2 Segmentačná metóda

Mikroskopové dátá makrofágov, s ktorými pracujeme v tejto práci, sú makrofágy priesvitného embrya zebričky pruhovanej (*lat. danio rerio*). Toto embryo bolo poranené a cytoplazmy makrofágov sú zafarbené zeleným svetielkujúcim proteínom (kaede) pre lepšiu viditeľnosť pod mikroskopom. Dátá makrofágov sú získané v časovom úseku 5 hodín s časovým krokom 4 minúty. Trojrozmerné obrazové dátá sú premietnuté do roviny za použitia maximálnej intenzity približne zo 70 rezov, z ktorých tak vzniknú dvojrozmerné obrazové dátá, ktoré budeme spracovať.

Hlavnou úlohou pri segmentácii makrofágov je rozlíšenie pozadia a objektu (makrofágu) na obrazových dátach. Táto úloha môže byť stážená kvôli prítomnosti šumu v dátach a faktu, že makrofágy majú nepravidelné tvary s meniacou sa intenzitou. Preto sme vybrali a skúmali niekoľko druhov prahovacích metód v snahe vylepšiť prahovanie v [3], ktoré sme skombinovali s metódou segmentácie subjektívnych plôch SUBSURF a aplikovali sme ich na testované dátá.

2.1 Globálne prahovanie

Úlohou globálnych prahovacích metód je nájsť jedinú optimálnu prahovú hodnotu q , v našom prípade šedo-tónových obrazových dát, ktorá zadefinuje každý pixel obrazu buď do popredia (ako objekt na dátach) alebo ako pozadie. Na základe jedinej hodnoty sú teda všetky pixely zatriedené do dvoch disjunktných množín C_0 a C_1 , kde množina C_0 obsahuje všetky pixle s hodnotami intenzity z množiny $(0, 1, \dots, q)$ a C_1 obsahuje všetky zvyšné pixle s intenzitou z množiny $(q + 1, \dots, K - 1)$. Platí teda

$$(u, v) \in \begin{cases} C_0 & \text{ak } I(u, v) \leq q \quad (\text{pozadie}) \\ C_1 & \text{ak } I(u, v) \geq q \quad (\text{objekt}) \end{cases}. \quad (1)$$

Treba si uvedomiť, že tieto hodnoty závisia od toho, či je pozadie bledé a objekt tmavý alebo naopak.

Mnohé z týchto metód sú založené na práci s histogramom a sú zvyčajne jednoduché a účinné, pretože pracujú s malým množstvom dát. V našom prípade sa jedná o 256 odtieňov sivej/šede. Dajú sa rozdeliť na 2 hlavné kategórie: štatistické metódy a také, ktoré sú založené na tvare.

2.1.1 Otsuho metóda

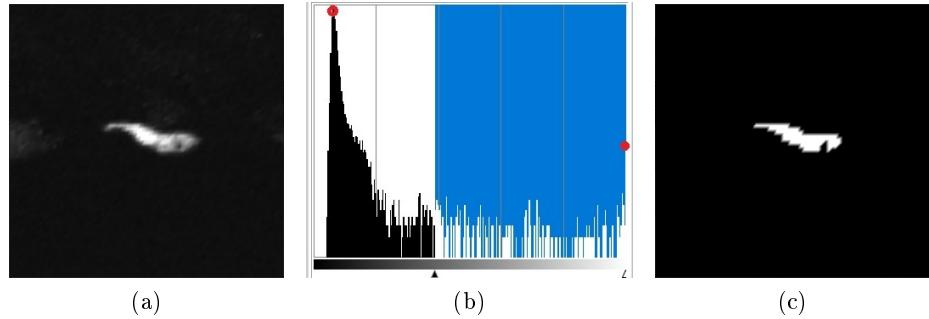
Otsuho metóda[7] patrí medzi automatické prahovacie metódy, ktorá rozdeľuje obrazové dátá na 2 rôzne triedy pomocou prahu q - na objekt a pozadie. Hlavnou myšlienkou tejto metódy je nájsť prah q taký, že výsledné distribúcie tried sú čo najlepšie oddelené, čo znamená, že (Obr.1):

- príslušné histogramy sú čo najužšie (majú čo najmenšiu varianciu),
- vrcholy histogramov pre pozadie a objekt sú od seba čo najďalej.

Na dosiahnutie prvej vlastnosti sa na výpočet prahu q používa veličina známa ako vnútrotriedna variancia (*eng. within class variance*), pričom sa hľadá jej minimum. Dá sa ukázať, že táto úloha môže byť zmenená na maximalizačnú úlohu tzv. medzi-triednej variancie (*eng.between class variance*), ktorá je výpočtovo menej náročná a súčasne zabezpečuje druhú vlastnosť.

Vnútro-triedna variacia (*eng. within-class variance*) Var_w pre dve triedy C_0 a C_1 je vážený priemer ich variancií V_0 a V_1 , kde váhy P_0 a P_1 sú dané relatívou početnosťou pixlov objektu resp. pozadia (ich percentuálnym zastúpením). Definuje sa ako:

$$Var_w = P_0 \cdot V_0 + P_1 \cdot V_1. \quad (2)$$



Obr. 1: Na obrázku *b* je prahová hodnota naznačená čiernym trojuholníkom, histogram pre pozadie je vľavo od nej, histogram pre objekt vpravo. Červené krúžky naznačujú vrcholy histogramu pozadia a objektu. Prahová hodnota je 99. Relatívna početnosť pixlov pozadia P_0 (percento/100) je 0,984, pixlov objektu P_1 je rovná 0,016. Priemerná hodnota pozadia je 17,664, objektu 181,76. Každá trieda má vlastnú varianciu V_0 a V_1 . $P_0 \cdot V_0 + P_1 \cdot V_1$ dáva tzv. medzi-triednu varianciu. Táto hodnota prahu 99 je podľa Otsuho metódy optimálna. Ak urobíme prahovanie inou hodnotou, napr. 50, ktoré lepšie obalí objekt, pokazí sa hlavne variancia objektu a vnútrotriedna variancia bude vyššia.

Na dosiahnutie druhej vlastnosti sa počítá *medzi-triedna variancia* Var_b (*eng. between-class variance*), ktorá meria vzialenosť priemerov tried μ_0 a μ_1 . Jej výpočet podľa definície je založený na výpočte priemerov pixlov pozadia μ_0 , pixlov objektu μ_1 a priemernej intenzite celého obrázku μ_I , ktorá však po úprave z výpočtu vypadne. Definuje sa ako:

$$Var_b = P_0 \cdot (\mu_0 - \mu_I)^2 + P_1 \cdot (\mu_1 - \mu_I)^2 \quad (3)$$

Tento výraz treba maximalizovať. Celková variancia obrázku Var_I je sumou vnútrotriednej a medzi-triednej variancie. Platí

$$Var_I = Var_w + Var_b.$$

Kedže Var_I je konštantná pre daný obrázok, na získanie optimálneho prahu q si môžeme vybrať, či chceme minimalizovať Var_w alebo maximalizovať Var_b . Výpočtovo jednoduchší je však druhý postup. Urobme nasledovnú úpravu. Kedže $\mu_I = P_0 \cdot \mu_0 + P_1 \cdot \mu_1$, tak môžeme napísat:

$$\begin{aligned} P_0(\mu_0 - \mu_I)^2 &= P_0(\mu_0 - P_0\mu_0 - P_1\mu_1)^2 \\ &= P_0((1 - P_0)\mu_0 - P_1\mu_1)^2 \\ &= P_0(P_1\mu_0 - P_1\mu_1)^2 \\ &= P_0P_1^2(\mu_0 - \mu_1)^2 \end{aligned} \quad (4)$$

Podobne platí

$$P_1(\mu_1 - \mu_I)^2 = P_0^2P_1(\mu_0 - \mu_1)^2. \quad (5)$$

Nakoniec

$$\begin{aligned} Var_b &= P_0P_1^2(\mu_0 - \mu_1)^2 + P_0^2P_1(\mu_0 - \mu_1)^2 \\ &= P_0P_1(P_1 + P_0)(\mu_0 - \mu_1)^2 \\ &= P_0P_1(\mu_0 - \mu_1)^2, \end{aligned} \quad (6)$$

kedže $P_1 + P_0 = 1$. Všetky členy s indexom 0 alebo 1 závisia na prahu q . Pri maximalizácii budeme postupne brať za q všetky možné hodnoty intenzity, pričom bude počítať iba priemery a počty pixlov pozadia a objektu, čo sa dá vykonať rýchlo iteráčne, čiže budeme počítať iba

štatistiku prvého rádu. Maximalizovať Var_b je teda rýchlejšie ako minimalizovať Var_w .

Priemery tried sa dajú jednoducho vypočítať z normalizovaného histogramu. Nech K je maximálna možná intenzita obrazu (v našom prípade 255), hodnoty normalizovaného histogramu budú vypočítané ako

$$p_i = \frac{h_i}{N}, \quad \sum_{i=0}^K p_i = 1, \quad (7)$$

kde i je konkrétna úroveň intenzity ($0 \leq i \leq K$), N je celkový počet pixlov v obraze a h_i predstavuje počet pixlov intenzity i v celom obraze. Ďalej je potrebné vypočítať stredné hodnoty μ_0 a μ_1 podľa

$$\mu_0 = \sum_{i=0}^q \frac{ip_i}{\omega_0(q)}, \quad \mu_1 = \sum_{i=q+1}^K \frac{ip_i}{\omega_1(q)}, \quad (8)$$

kde $\omega_0(q)$ a $\omega_1(q)$ sú sumy definované ako

$$\omega_0(q) = \sum_{i=0}^q p_i, \quad \omega_1(q) = \sum_{i=q+1}^K p_i. \quad (9)$$

Kvalita (vhodnosť) prahovania sa dá merať pomerom

$$\nu = \frac{Var_b(q_{max})}{Var_I} \quad (10)$$

ktorý nadobúda hodnoty z intervalu $[0,1]$ a je invariantný na zmenu kontrastu. Čím vyššia je hodnota, tým lepšie je prahovanie. Pre dátu z obr.1 je táto hodnota 0.82.

Otsuho metóda dokáže dobre rozlíšiť dátu s makrofágmi v prípade, že dátu neobsahujú výrazný šum aj v prípade keď sa na dátach nachádzajú tenké časti alebo sú zložito tvarované. Avšak ak je intenzita šumu pozadia porovnatelná s intenzitou objektu, táto metóda môže spôsobiť rozdelenie objektu a stratiť niektoré jeho časti, keďže je do úvahy braná len intenzita obrazu.

2.1.2 Prahovanie pomocou maximálnej entropie

Entropia je dôležitým pojmom v teórii informácií a najmä pri kompresii dát. Je to štatistická miera, ktorá kvantifikuje priemerné množstvo informácií obsiahnutých v "správe" obsahujúcej stochasticky generované dátu. Entropia je definovaná ako

$$H(I) = - \sum_{g=0}^K p(g) \log_b(p(g)), \quad (11)$$

kde g je intenzita, $p(g)$ je pravdepodobnosť tejto intenzity v normalizovanom histograme a b je zvolený logaritmický základ, väčšinou $b = 10$, $b = 2$ alebo $b = e$. Hodnota entropie H bude vždy nadobúdať kladnú hodnotu, pretože argument logaritmu sú pravdepodobnosti, ktoré sú z intervalu $(0, 1)$, z čoho vyplýva že hodnota logaritmu bude vždy záporná.

Na hľadanie maximálnej hodnoty entropie potrebujeme definovať entropie H_0 a H_1 pre každú triedu C_0 a C_1 nasledovne

$$H_0(q) = - \sum_{i=0}^g p(i) \log_b(p(i)), \quad (12)$$

$$H_1(q) = - \sum_{i=q+1}^K p(i) \log_b(p(i)) \quad (13)$$

a celková entropia je ich súčet. Po úpravách môžme napísat [5, 8]

$$H_0(q) = -\frac{1}{P_0(q)} S_0(q) + \log(P_0(q)) \quad (14)$$

$$H_1(q) = -\frac{1}{1-P_0(q)} S_1(q) + \log(1-P_0(q)), \quad (15)$$

kde P_0 predstavuje kumulatívnu pravdepodobnosť a S_0, S_1 (uvedené ďalej) sú vopred vyrátané sumy $-\sum_{i=0}^q p(i) \log_b(p(i))$ a $-\sum_{i=q+1}^K p(i) \log_b(p(i))$.

Celková entropia pre daný prah q je daná ako

$$H(q) = H_0(q) + H_1(q); \quad (16)$$

Kumulatívna pravdepodobnosť P_0 je definovaná ako

$$P_0(q) = \begin{cases} p(0) & \text{pre } q = 0 \\ P_0(q-1) + p(q) & \text{pre } 0 < q < K, \end{cases} \quad (17)$$

a sumačné podmienky S_0, S_1 sú predpočítané a definované ako

$$S_0(q) = \begin{cases} p(0).log(p(0)) & \text{pre } q = 0 \\ S_0(q-1) + p(q)log(p(q)) & \text{pre } 0 < q < K \end{cases} \quad (18)$$

$$S_1(q) = \begin{cases} 0 & \text{pre } q = L-1 \\ S_0(q+1) + p(q+1)log(p(q+1)) & \text{pre } 0 \leq q < K-1 \end{cases} \quad (19)$$

Táto metóda je jednoduchá a účinná, pretože závisí len od histogarmu obrazu. Entropia ako kritérium na voľbu prahu v obrazových dátach má dlhú tradíciu a navrhnutých bolo viacero metód. Vyššie uvedená metóda je jednou zo starších metód a bola navrhnutá matematikom J. N. Kapurom. Pre biologické dáta väčšinou bola nadmnožinou Otsuho prahovania, ale v mnohých prípadoch nepokryla celý objekt (porovnajte obrázky 15 a 18).

2.2 Lokálne adaptívne prahovanie

Lokálne adaptívne prahovanie namiesto jednej prahovej hodnoty pre celý obraz, používa adaptívne prahovanie, ktoré určuje meniacu sa prahovú hodnotu $Q(u, v)$ pre každú polohu obrazu. Tieto hodnoty zodpovedajú každému pixlu $I(u, v)$ zodpovedajúcemu danému obrazu. Nasledujúce metódy sa líšia iba s ohľadom na to, akým spôsobom sú získané prahy Q zo vstupného obrázku.

2.2.1 Bernsenova metóda

Táto metóda, ktorá určuje prah dynamicky pre každú polohu na obrazových dátach (u, v) , je založená na minimálnej a maximálnej intenzite nachádzajúcej sa v okolí $R(u, v)$. Ak

$$\begin{aligned} I_{min}(u, v) &= \min_{(i,j) \in R(u,v)} I(i, j), \\ I_{max}(u, v) &= \max_{(i,j) \in R(u,v)} I(i, j), \end{aligned} \quad (20)$$

sú minimálnou a maximálnou hodnotou intenzity, na nejakom fixne danom okolí R so stredom na pozícii (u, v) . Prahovú hodnotu dostaneme pomocou aritmetického priemeru nájdenej minima a maxima daného okolia

$$Q(u, v) \leftarrow \frac{I_{min}(u, v) + I_{max}(u, v)}{2} \quad (21)$$

Takéto nastavenie sa vykoná ak lokálny kontrast $c(u, v) = I_{max}(u, v) - I_{min}(u, v)$ je väčší ako preddefinovaná hodnota c_{min} . Ak $c(u, v) < c_{min}$, tak predpokladáme, že pixle patria do tej istej triedy a sú automaticky priradené do pozadia alebo objektu, napríklad porovnaním s intenzitou. Pre tento algoritmus je dôležité akú nastavíme šírku (polomer) okna - pre biologické dátá je to malá hodnota. V prípade šumu treba vhodne zvoliť c_{min} .

2.2.2 Niblackova metóda

Prah $Q(u, v)$ pri Niblackovej metóde sa mení ako funkcia lokálneho priemeru intenzít $\mu_R(u, v)$ a smerodajnej odchýlky $\sigma_R(u, v)$, v tvare

$$Q(u, v) = \mu_R(u, v) + \kappa\sigma_R(u, v). \quad (22)$$

Lokálny prah je určený pridaním konštanty $\kappa \geq 0$ k smerodajnej odchýlke $\sigma_R(u, v)$ a lokálneho priemeru $\mu_R(u, v)$. Lokálne hodnoty smerodajnej odchýlky $\sigma_R(u, v)$ dostaneme ako

$$\sigma_R(u, v) = \frac{1}{N} \sum_{(i,j) \in R(u,v)} (I(i, j) - \overline{I(i, j)})^2, \quad (23)$$

kde R označuje fixne dané okolie, so stredom v (u, v) , N je počet prvkov nachádzajúcich sa v okolí R , $I(i, j)$ sú označené pixle obrazu nachádzajúce sa v okolí R a $\overline{I(i, j)}$ je priemer pixlov patriacich okoliu R . Po úprave dostaneme tvar

$$\sigma_R(u, v) = \overline{I(i, j)^2} - (\overline{I(i, j)})^2, \quad (24)$$

kde $\overline{I(i, j)}$ je priemerom intenzít z fixne daného okolia R a $\overline{I(i, j)^2}$ je priemerom druhých mocnín intenzít z fixne daného okolia R . Tieto priemery budeme približne počítať pomocou rovnice vedenia tepla, pričom šírke okna bude zodpovedať čas.

Budeme hľadať funkciu $u(x, t)$, kde $x \in \Omega$, $t \in [0, T]$ a PDR je definovaná v tvare

$$\frac{\partial u(x, t)}{\partial t} = \Delta u(x, t), \quad (25)$$

s Neumanovými okrajovými podmienkami na hranici $\partial\Omega$ v tvare

$$\frac{\partial u(x, t)}{\partial \vec{n}} = 0, \quad (26)$$

kde \vec{n} je jednotková vonkajšia normála ku hranici $\partial\Omega$ a počiatočnou podmienkou

$$u(x, 0) = u^0(x), \quad (27)$$

ktorá je určená počiatočnými obrazovými dátami.

Výpočet pre lokálny prah je definovaný na oblasti R so stredom v (u, v) . Polomer oblasti R by mala byť čo najväčšia, aspoň tak veľká ako štruktúra, ktorú sa vyprahovaním snažíme získať, ale dostatočne malá na zachytenie zmien (nerovností) pozadia.

Jeden z problémov, ktorý môže nastať pre malé hodnoty smerodajnej odchýlky $\sigma_R(u, v)$ (získané na oblastiach v obrazových dátach s takmer konštantnou intenzitou), prah bude mať hodnotu blízku lokálnemu priemeru, čo spôsobí, že segmentácia je pomerne citlivá na nízku amplitúdu šumu ("ghosting"). Pomocou jednoduchej modifikácie rovnice (22) pridaním konštanty d , ktorá zabezpečí minimálnu vzdialenosť od priemeru v tvare

$$Q(u, v) = \mu_R(u, v) + \kappa\sigma_R(u, v) + d, \quad (28)$$

kde $d \geq 0$. V našom prípade, sme parameter κ nastavili na $\kappa = 0.18$ a pre konštantu d sme použili $d = 20$. Takto nastavené parametre dávali v našom prípade veľmi dobré výsledky pre takmer všetky makrofágy. Zväčšovaním času sa vysegmentovaný objekt zväčšuje. Výpočet s menším časom sa dá použiť na vylepšenie výpočtu Perona-Malikových koeficientov v modeli SUBSURF, pomocou výpočtu s väčším časom sa dá k nemu získať počiatočná podmienka.

Sauvolova prahovacia metóda je vylepšením Niblackovej metódy. Prah $Q(u, v)$ je definovaný v tvare

$$Q(u, v) = \mu_R(u, v)[1 + \kappa(\frac{\sigma_R(u, v)}{\sigma_{max}} - 1)], \quad (29)$$

kde parameter $\kappa \geq 0$, σ_{max} je dynamickým rozsahom pre štandardné odchýlky. Približne hodnoty priemeru intenzít $\mu_R(u, v)$ a smerodajnej odchýlky $\sigma_R(u, v)$ na fixne danej oblasti R sme dostali z rovnice vedenia tepla, tak ako je popísaná vyššie. Parametre metódy boli v našom prípade zvolené nasledovne $\kappa = 0.18$ a $\sigma_{max} = 128$.

2.2.3 Hybridné metódy

Pri hybridnej Niblackovej a Bernsenovej metóde[6], uvažujeme lokálny kontrast $c(u, v)$ definovaný v Bernsenovej metóde v sekciu 2.2.1 a na výpočet lokálneho prahu $Q(u, v)$ použijeme rovnicu (28), z Niblackovej metódy popísanú v sekciu 2.2.2. Lokálny kontrast $c(u, v)$ aj prah $Q(u, v)$ sú definované na oblasti R .

Pri druhej hybridnej metóde, použijeme kombináciu Bernsenovej metódy a Sauvolovej metódy[6], podobne ako pri predchádzajúcom prípade lokálny kontrast $c(u, v)$ vypočítame z Bernsenovej metódy a na výpočet lokálneho prahu $Q(u, v)$ použijeme rovnicu (29), zo Sauvolovej metódy. Lokálny kontrast $c(u, v)$ aj prah $Q(u, v)$ sú definované na oblasti R .

2.3 Metóda subjektívnych plôch (SUBSURF)

Metóda subjektívnych plôch (SUBSURF) je vysoko účinnou segmentačnou metódou, ktorá dokáže účinne nájsť chýbajúce hranice objektu alebo odstrániť šum z pozadia. Avšak kvalita výsledku segmentácie závisí od voľby počiatočnej podmienky. Keďže väčšina makrofágov má zložitý tvar, môžeme predpokladať, že metóda SUBSURF by nedávala správne výsledky ak by nebola vhodne zvolená počiatočná podmienka. Za počiatočnú podmienku zvolíme dátu, na ktoré bola aplikovaná niektorá z vyššie uvedených prahovacích metód.

Použitá metóda má tvar

$$u_t = \sqrt{\varepsilon^2 + |\nabla u|^2} \nabla \cdot (g \frac{\nabla u}{\sqrt{\varepsilon^2 + |\nabla u|^2}}), \quad (30)$$

kde ε je modelovacím parametrom a u je vyvíjačuca sa level funkcia. Funkcia g reprezentuje takzvaný hranový detektor a má tvar

$$g(s) = \frac{1}{1 + ks^2}, k > 0, \quad (31)$$

kde $s = |\nabla G_\sigma * I^0|$. Zhľadený gradient $\nabla G_\sigma * I^0$ získame napríklad aplikovaním jedného kroku vedenia tepla.

2.3.1 Modifikácia metódy

Hranový detektor pôvodnej SUBSURF metódy (31) boli modifikované dvomi rôznymi spôsobmi na získanie lepších výsledkov.

V prvom prípade sme na získanie hranového detektora $g1(s)$ použili priemer z pôvodných dát a dát, na ktoré sme použili niektorú z globálnych alebo lokálnych adaptívnych prahovacích metód nasledovne

$$g1(s) = \frac{1}{1 + k|\nabla \frac{u_o + u_t}{2}|^2}, \quad (32)$$

kde u_o označujú pôvodné dátá a u_t sú vyprahované dátá.

Pri druhom spôsobe modifikácie hranového detektora $g2(s)$ [3], je použitá kombinácia gradientov predvyhľadených pôvodných dát a predvyhľadených dát, na ktoré sme použili niektorú z globálnych alebo lokálnych adaptívnych prahovacích metód ako

$$g2(s) = \left(\frac{C_o}{1 + k|\nabla u_o|^2} + \frac{C_t}{1 + k|\nabla u_t|^2} \right), \quad (33)$$

kde u_o označujú pôvodné dátá, u_t sú vyprahované dátá, C_o a C_t sú koeficienty hranového detektora. Koeficienty hranového detektora mali hodnoty $C_o = 0.2$ a $C_t = 0.8$.

3 Numerické schémy

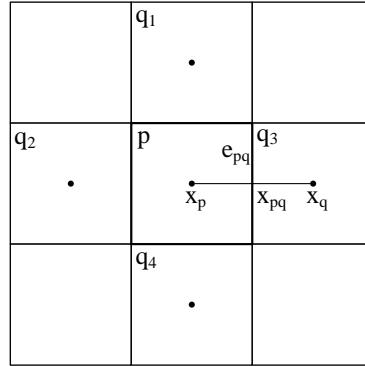
Pri vytváraní numerickej schémy sme pre časovú diskretizáciu použili implicitnú a semi–implicitnú schému a pri priestorovej diskretizácii bola použitá metóda konečných objemov.

3.1 Implicitná metóda pre rovniciu vedenia tepla

Pri implicitnej metóde, budeme časovú deriváciu aproximovať pomocou spätej diferencie a pravú stranu rovnice berieme v novom časovom kroku n . Má tvar

$$\frac{u^n - u^{n-1}}{\tau} = \Delta u^n = \nabla \cdot (\nabla u^n) \quad (34)$$

a pre ľubovoľnú veľkosť časového kroku je bezpodmienečne stabilná.



Obr. 2: Detail siete spolu s označením. V strede sa nachádza pixel p , susedné pixle sú označené ako q_i , kde $q_i = 1, 2, 3, 4$, hrany medzi pixlami označíme ako e_{pq} , x_p a x_q sú body.

Pre pixelovú sieť budeme používať označenie uvedené na obr. 2. Rovnicu (34) integrujeme cez konečný objem p a dostaneme rovnicu v tvare

$$\int_p \frac{u^n - u^{n-1}}{\tau} dx = \int_p \nabla \cdot (\nabla u^n) dx. \quad (35)$$

Na pravú stranu rovnice aplikujeme Greenovu vetu a dostaneme

$$\int_p \nabla \cdot (\nabla u^n) dx = \int_{\partial p} \nabla u^n \cdot \vec{n}_p dS, \quad (36)$$

kde \vec{n}_p je jednotkovou vonkajšou normálou ku hranici konečného objemu p . Pretože platí

$$\int_{\partial p} \nabla u^n \cdot \vec{n}_p dS = \sum_{q \in N(p)} \int_{e_{pq}} \nabla u^n \cdot \vec{n}_{pq} dS, \quad (37)$$

vieme napísť slabú konečno-objemovú formuláciu úlohy

$$\int_p \frac{u^n - u^{n-1}}{\tau} dx = \sum_{q \in N(p)} \int_{e_{pq}} \nabla u^n \cdot \vec{n}_{pq} dS = \sum_{q \in N(p)} \int_{e_{pq}} \frac{\partial u^n}{\partial \vec{n}_{pq}} dS, \quad (38)$$

Riešenie v MKO chápeme ako po častiach konštantnú funkciu na konečných objemoch p , označíme u_p^n . Preto vieme ľavú stranu rovnice napísť ako

$$\int_p \frac{u^n - u^{n-1}}{\tau} dx = \frac{u_q^n - u_p^n}{\tau} m(p). \quad (39)$$

Pravá strana reprezentuje tok (flux) cez hranicu e_{pq} v smere normály \vec{n}_{pq} . Člen ľavej strany $\nabla u^n \cdot \vec{n}_{pq}$ aproximujeme na hrane e_{pq} konečnou diferenciou v bode x_{pq} nasledovne

$$\nabla u^n \cdot \vec{n}_{pq} \approx \frac{u_q^n - u_p^n}{d_{pq}}. \quad (40)$$

V časovom kroku n aproximáciu x_{pq} použijeme na celej hrane e_{qp} a dostaneme

$$\sum_{q \in N(p)} \int_{e_{pq}} \nabla u^n \cdot \vec{n}_{pq} dS \approx \sum_{q \in N(p)} m(e_{pq}) \frac{u_q^n - u_p^n}{d_{pq}}. \quad (41)$$

Po dosadení dostávame implicitnú schému

$$\frac{m(p)(u_p^n - u_p^{n-1})}{\tau} = \sum_{q \in N(p)} m(e_{pq}) \frac{(u_q^n - u_p^n)}{d_{pq}} \quad (42)$$

kde platia vzťahy $m(p) = h^2, m(e_{pq}) = h, d_{pq} = h$. Po dosadení dostaneme lineárny rovnicový systém

$$u_p^n + \frac{\tau}{h^2} \sum_{q \in N(p)} (u_q^n - u_p^n) = u_p^{n-1}. \quad (43)$$

Pri implicitnej schéme platí bezpodmienečná stabilita (diskrétny princíp minima a maxima), čo znamená, že ak pre ľubovoľnú voľbu priestorového kroku h a časového kroku τ , platí $u_{min} \leq u_p^{n-1} \leq u_{max}$ potom platí aj pre $u_{min} \leq u_p^n \leq u_{max}$. Riešenie lineárneho systému upravíme do nasledovného tvaru

$$(1 + \frac{\tau}{h^2} \sum_{q \in N(p)} 1) u_p^n - \frac{\tau}{h^2} \sum_{q \in N(p)} u_q^n = u_p^{n-1}. \quad (44)$$

Rovnicu (44) dostaneme na každom konečnom objeme p . Následne schému riešime pomocou super-relaxačnej metódy (eng. Successive Over-Relaxation method - SOR).

3.2 Numerická schéma pre SUBSURF

V prvom rade urobíme úpravu do divergentného tvaru, tak že rovnicu (30) vydelíme členom $\sqrt{\varepsilon^2 + |\nabla u|^2}$ a dostaneme

$$\frac{1}{|\nabla u^{n-1}|_\varepsilon} \cdot \frac{u^n - u^{n-1}}{\tau} = \nabla \cdot (g^0 \frac{\nabla u^n}{|\nabla u^{n-1}|_\varepsilon}), \quad (45)$$

kde $g^0 = g(|\nabla G_\sigma * I^0|)$ a $|\nabla G_\sigma * I^0|$ predstavuje zhľadený gradient. Teraz spravíme priestorovú diskretizáciu pomocou metód konečných objemov, čo znamená, že zintegrujeme rovnicu (45) cez konečný objem p a dostaneme

$$\int_p \frac{1}{|\nabla u^{n-1}|_\varepsilon} \cdot \frac{u^n - u^{n-1}}{\tau} dx = \int_p \nabla \cdot (g^0 \frac{\nabla u^n}{|\nabla u^{n-1}|_\varepsilon}) dx, \quad (46)$$

Následne na pravú stranu rovnice aplikujeme Greenovu vetu

$$\int_p \frac{1}{|\nabla u^{n-1}|_\varepsilon} \cdot \frac{u_p^n - u_p^{n-1}}{\tau} dx = \int_{\partial p} g^0 \frac{\nabla u^n}{|\nabla u_p^{n-1}|_\varepsilon} \vec{n}_{pq} dS, \quad (47)$$

Derivácia v smere vonkajšej normály ku konečnému objemu p je v rovnici (47) reprezentovaná ako $\nabla u^n \cdot \vec{n}_{pq}$. Na ľavej strane rovnice (47) budeme uvažovať konštantnú reprezentáciu riešenia a jeho gradientu. Normálovú deriváciu z pravej strany rovnice nahradíme konečnou diferenciou hodnotami, ktoré reprezentujú hodnoty na hranách konečného objemu p . Nasledovne

$$\frac{m(p)}{|\nabla u_p^{n-1}|_\varepsilon} \frac{u_p^n - u_p^{n-1}}{\tau} = \sum_{q \in N(p)} \int_{e_{pq}} g_{pq}^0 \frac{u_q^n - u_p^n}{d_{pq}} \cdot \frac{1}{|\nabla u_{pq}^{n-1}|_\varepsilon} ds. \quad (48)$$

Vyčíslením integrálu v časovom kroku n aproximáciu x_{pq} použijeme na celej hrane e_{qp} a dostaneme

$$\frac{m(p)}{|\nabla u_p^{n-1}|_\varepsilon} \frac{u_p^n - u_p^{n-1}}{\tau} = \sum_{q \in N(p)} g_{pq}^0 \frac{u_q^n - u_p^n}{d_{pq}} \cdot \frac{m(e_{pq})}{|\nabla u_{pq}^{n-1}|_\varepsilon}. \quad (49)$$

Ked'že ide o pixelovú sieť s hranu h , dostaneme tvar

$$\frac{h^2}{\tau |\nabla u_p^{n-1}|_\varepsilon} u_p^n + \sum_{q \in N(p)} \frac{g_{pq}^0}{|\nabla u_{pq}^{n-1}|_\varepsilon} (u_q^n - u_p^n) = \frac{h^2}{\tau |\nabla u_p^{n-1}|_\varepsilon} u_p^{n-1}. \quad (50)$$

Z tvaru (50), dostaneme rovnicu pre každý pixel

$$a_p^{n-1} u_p^n - \sum_{q \in N(p)} a_{pq}^{n-1} u_q^n = b_p^{n-1} u_p^{n-1}, \quad (51)$$

kde $a_{pq}^{n-1}, a_p^{n-1}, b_p^{n-1}$ označujú

$$\begin{aligned} a_{pq}^{n-1} &= \frac{g_{pq}^0}{|\nabla u_{pq}^{n-1}|_\varepsilon}, \\ a_p^{n-1} &= \frac{h^2}{\tau |\nabla u_p^{n-1}|_\varepsilon} + \sum_{q \in N(p)} a_{pq}^{n-1}, \\ b_p^{n-1} &= \frac{h^2}{\tau |\nabla u_p^{n-1}|_\varepsilon}, \end{aligned} \quad (52)$$

a po dosadení okrajových podmienok dostávame sústavu lineárnych rovníc riešenú v novom časovom kroku u^n . V našom prípade boli na získanie hranového detektora g_{pq}^0 uvažované dva rôzne prípady popísané v sekcií 2.3.1.

4 Softvér

Na implementáciu a vytvorenie prostredia, bol zvolený objektovo orientovaný prístup jazyka C++, spolu s knižnicami Qt [2], ktoré obsahujú veľa na implementovaných tried a boli užitočným nástrojom pri vytváraní užívateľského prostredia a VTK [1] knižnicami, ktoré slúžia na zobrazovanie a manipuláciu s dátami, s ktorými program pracuje.

4.1 Qt

Užívateľské rozhranie je vytvorené pomocou Qt knižíc, ktoré sú jedným z najpoužívanejších cross-platformových frameworkov na vytváranie užívateľského prostredia (GUI). Majú aj veľa predprogramovaných knižíc s rôznymi funkciami, ktoré programátorovi uľahčia prácu. Sú naimplementované v jazyku C++.

Najčastejšie využívané Qt triedy v projekte:

- **QMdiArea**

Táto trieda zohráva jednu z najdôležitejších funkcií v programe. Funkcie tejto triedy fungujú v podstate ako správca okien pre MDI okná, čo v našom prípade znamená, že umožňuje vytvárať podokná pomocou triedy, v ktorých sa v programe nachádzajú ďalšie Qt triedy slúžiace na vykreslovanie 2D a 3D dát, s ktorými program pracuje. V programe je použité kaskádové usporiadanie takýchto podokien, čo znamená že vykreslovacie okná sa môžu navzájom prekrývať, dajú sa minimalizovať/maximalizovať vrámcí hlavného okna a zavrieť.

- **QScrollArea**

QScrollArea sa nachádza v každom podokne widgetu QMdiArea. Zabezpečuje možnosť priblížiť/oddialiť a posúvať vizualizované dátu. Tiež sa nachádza aj v častiach programu, kde bolo potrebné aplikovať ScrollBar.

- **QVTKOpenGLNativeWidget**

Widget tejto triedy sa nachádza v každej QScrollArea a umožňuje samotné vykreslenie 2D/3D modelov za pomoci VTK knižníc.

- **QDockWidget**

Objektmi tejto triedy sú widgety, ktoré v našom programe obsahujú všetky informácie a nastavenia súvisiace s dátami. Každý logický celok má vlastný panel (*eng. dock*), ktorý sa dá vrámcí okna premiestňovať, ukotvovať buď na ľavej alebo na pravej strane okna a v prípade, že sa prekrývajú vytvorí sa z nich viacero záložiek. Taktiež sa dajú v prípade potreby minimalizovať.

- **QTreeWidget**

Dáta, či už pôvodné, vyprahované alebo vysegmentované pomocou programu, sa nachádzajú v zozname, z ktorého sa dá zvoliť, ktoré dátu budú vykreslené, resp. s ktorými dátami bude program pracovať. QTreeWidget bol použitý aby sme vykreslované dátu vedeli zadať medzi 2D alebo 3D dátu.

- **QVector**

Trieda QVector, je šablónovou triedou, ktorá definuje dynamické polia. Ukladá premenné do susedných miesta v pamäti a poskytuje rýchly indexový prístup. Je použitá v prípadoch keď nie je potrebné odstraňovať prvky zvnútra QVectora.

- **QFile, QFileDialog**

Tieto triedy slúžia v programe na otváranie, načítavanie, ukladanie a manipuláciu s dátami.

- **QHelp**

Pomocou tejto triedy bol vytvorený pomocník vytvorením niekoľkých jednoduchých `html` súborov popisujúcich základnú funkciu programu.

4.2 VTK

Visualization Toolkit (VTK) sú voľne dostupnými knižnicami, ktoré v programe slúžia na zobrazovanie a interakciu s 2D aj 3D dátami. Spomenieme niektoré knižnice, ktoré hrajú dôležitú úlohu pri našej implementácii.

- **vtkSmartPointer**

Táto šablónová trieda, slúži ako pointer pre VTK triedy. Jeho úlohou je zlepšiť manažment s pamäťou, čo znamená, že v prípade ak sú dátá mimo rozsahu alebo sa nikde nepoužívajú tak budú automaticky odstránené.

- **vtkTIFFReader**

Táto trieda je použitá pri načítavaní súborov typu TIFF.

- **vtkPoints**

Je triedou reprezentujúcou zoskupenie trojíc (x, y, z) 3D bodov a manipuláciou s uloženými bodmi.

- **vtkPolygon**

Uľahčuje vytváranie buniek n -stranného mnohouholníka v rovine. V našom prípade ide o štvoruholníky, znázorňujúce diskrétnu sieť. Každý štvoruholník reprezentuje jeden pixel načítaných obrazových dát.

- **vtkCellArray**

Objekty triedy `vtkCellArray` zabezpečujú prepojenie samostatných buniek rôznych typov, v našej implementácii ide o štvorcové (`vtkPolygon`) bunky. Štruktúra tejto triedy je reprezentovaná celočíselným poľom so štruktúrou v tvare: $(n, id_1, id_2, \dots, id_n, n, id_1, id_2, \dots, id_n, \dots)$, kde n je počet bodov, nachádzajúcich sa v bunke a id je index z pridruženého zoznamu bodov.

- **vtkPolyData**

V objektoch tejto triedy sú zadefinované vykreslované dátá, či už sa jedná o 2D alebo 3D dátá. V tejto triede môžu byť uložené informácie o tom akým spôsobom budú dátá reprezentované - geometrické informácie o štruktúre vykreslovaných dát. Takýmito informáciami môžu byť napríklad body, bunky, vektory, čiary, zafarbenie, polygonálne alebo trojuholníkové pásy.

- **vtkColorTransferFunction**

Je to trieda, ktorá nám slúži na definíciu farebných prechodov cez farebné modely RGB alebo HSV v počastiach spojitorom priestore závislom od hodnoty na súradnici z .

- **vtkPolyDataMapper**

Je triedou, zabezpečujúcou tzv. 'namapovanie' čo znamená, že zadefinuje, vlastnosti polydát definovaných ako objekt triedy `vtkPolyData`, ktoré sú potrebné pri následnom vykresení.

- **vtkActor**

Táto trieda reprezentuje, geometriu a vlastnosti vykreslovaných dát na zobrazovanej scéne. Odkazuje na geometriu uloženú ako objekt triedy na `vtkPolyDataMapper` a dedí funkcie súvisiace s pozíciou a orientáciou vykreslovaných dát. Tieto informácie sú uložené v transformačnej matici o veľkosti 4×4 , ktorá zabezpečuje rotácie vo všetkých smeroch (x, y, z) , škálovanie objektov atď..

- **vtkRenderer**

Trieda zabezpečujúca samotné vykreslovanie dát. Stará sa o prevod geometrie, špecifikuje svetelné podmienky a orientáciu kamery. Tiež vykonáva transformáciu súradníč medzi svetovými súradnicami, súradnicami zobrazenia (t.j. súradnicový systém počítačovej grafiky) a súradnicami displeja (t.j. súradnice obrazovky displeja).

- **vtkCutter**

Trieda **vtkCutter**, je filtrovacou triedou, ktorá pomocou implicitnej funkcie zabezpečuje vykreslovanie, viacerých alebo jedného rezu cez trojdimenzionálne dátá. Vo VTK na rezanie (eng. cutting) znamená zredukovanie dimenzie o jedna. V našom prípade išlo o 3D dátá, ktoré sme rezali pomocou roviny (**vtkPlane**).

- **vtkCubeAxesActor**

Pomocou tejto triedy, sú zobrazované hranice vykreslovaných dát spolu s označeniami osí a hodnotami nachádzajúcimi sa v smeroch x, y, z - výška, šírka a hĺbka.

- **vtkInteractorStyleJoystickCamera, vtkInteractorStyleImage**

Tieto dve triedy umožňujú prepínanie interakcie objektu s prostredím, v našom prípade išlo o odobranie možnosti rotácie pri 2D obrazových dátach.

- **vtkXMLPolyDataWriter**

Extrahuje dátá typu **vtkPolydata** do súboru. Štandardný formát vytvoreného súboru je **.vtpl**.

4.3 Rozdelenie projektu do tried

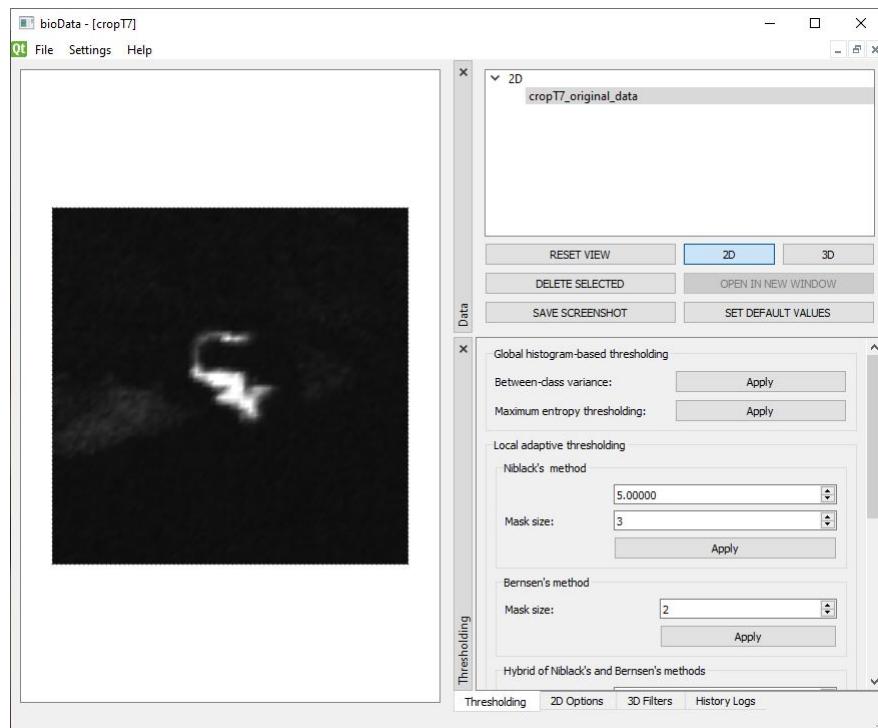
Projekt je rozdelený do viacerých súborov (tryed) dodržiavajúcej zásady objektovo-orientovaného programovania, z ktorých každá zabezpečuje funkcionality inej časti programu. Nasledovne:

- Trieda **bioData**, spája všetky tryedy a definuje užívateľské rozhranie pomocou **Qt** knižníca. Zabezpečuje prepojenie užívateľského prostredia, s naimplementovanými segmentačnými metódami, vykreslovacou plochou a zvyšnou funkcionalistou nachádzajúcou sa vo zvyšných triedach.
- **Source** zabezpečuje načítavanie dát zo súboru, následne sa tam vytvárajú sa dátá typu **vtkPolydata**, zadefinovaním bodov zo súboru, štvorcovej siete a definovaním farby/farebných prechodov. Tieto dátá sú následne zobrazené pomocou tryedy **viewerWidget**. Taktiež obsahuje funkcie na uloženie dát v rôznych formátoch.
- Trieda **filters**, obsahuje všetky funkcie, ktoré pracujú s načítanými dátami. Nachádzajú sa tam všetky prahovacie a segmentačné metódy a funkcie súvisiace s nimi.
- **viewerWidget** zabezpečuje vytvorenie plochy na vykreslovanie, vykreslovanie samotných 2D aj 3D dát, spolu s osami, izočiarami spolu s následnou aktualizáciou.
- Trieda **subWin** ukladá, hodnoty nastavené v užívateľskom rozhraní pre každé otvorené okno osobitne. Pri prepínaní medzi otvorenými oknami, sa aktualizujú všetky informácie nachádzajúce sa v bočnom paneli a v záložkách.
- Trieda **help** obsahuje funkcie prepájajúce vytvorenú dokumentáciu v **html** súboroch s užívateľským prostredím.

4.4 Grafické užívateľské rozhranie

V tejto sekcií sa oboznámime s vizuálnou stránkou vytvoreného softvéru a popíšeme funkcionality. Pri tvorbe programu sme sa zamerali na to aby bol čo najjednoduchší a vedel ho ovládať aj niekto, kto sa do danej problematiky až tak do hĺbky nevyzná. Grafické rozhranie(GUI) je vytvorené pomocou **Qt** knižníc.

Pri otvorení programu sa zobrazí okno, v ktorom sa nachádza horná lišta s položkami **File**, **Settings** a **Help** zvyšok bude načítaný až po načítaní súboru. V každej z týchto položiek sa nachádzajú ďalšie možnosti. Položka **File** obsahuje možnosti: **Open**, **Save**, **Close Files** a **Close**.



Obr. 3: Grafické rozhranie programu po načítaní vybraných dát.

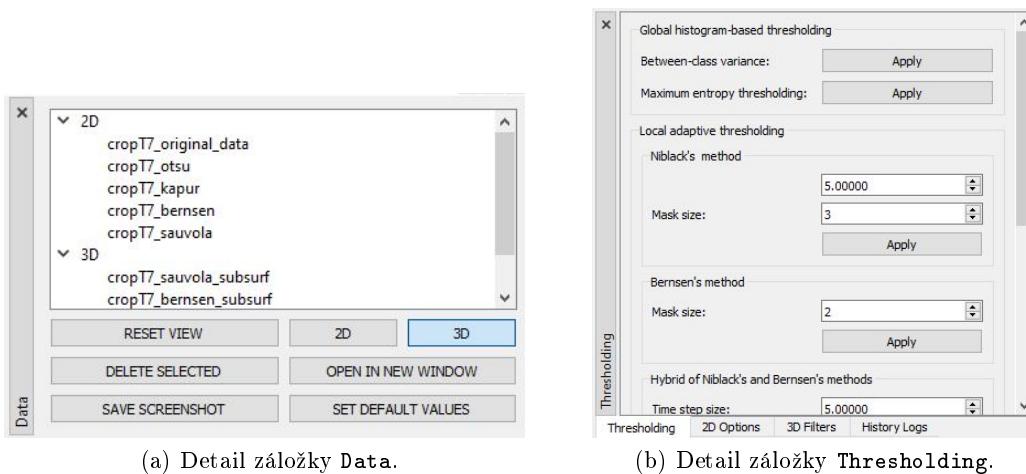
Po zvolení možnosti **Open**, sa otvorí dialógové okno, z ktorého sa dajú vybrať obrazové súbory typu .pgm (portable gray map) alebo .tif (TIFF). Po zvolení súboru sa zobrazí užívateľské prostredie aj so zvolenými dátami obr. 3. Na ľavej strane sú zobrazené načítané dátá. Na pravej strane, môžeme vidieť bočný panel rozdelený na dve časti, v ktorých sa nachádza zoznam spracovávaných dát v programe pre 2D aj 3D dátá, možnosti pre 2D a 3D dátá spolu so všetkými implementovanými filtrovmi, prahovacími metódami a možnosťami pre zobrazenie uložených vo viacerých záložkách.

Prvá časť panelu má názov *Data* obr. 4a obsahuje zoznam 2D a 3D dát, s ktorými sa doteraz pracovalo. Umožňuje spätné načítanie dát nachádzajúcich sa v zozname. Tiež obsahuje tlačidlá:

- **RESET VIEW** - vykreslovaný objekt, prekreslí na stred vykreslovacej plochy.
- **2D** - v prípade, že v zozname sú zvolené 2D dátá, odoberie možnosť rotácie a je vysvetlené.
- **3D** - v prípade, že v zozname sú zvolené 3D dátá, tak sa s nimi dá adekvátnie manipulovať a tlačidlo je vysvetlené.

- **DELETE SELECTED** - vymaže zvolené dátá zo zoznamu a v prípade, že sa jedná o posledné dátá v zozname, zavrie vykresľovacie okno spolu s bočným panelom.
- **OPEN IN NEW WINDOW** - otvorí 2D dátá zvolené v zozname v novom podokne.
- **SAVE SCREENSHOT** - uloží práve zobrazené dátá z vykresľovacej plochy ako obrazové dátá vo formáte *.png*
- **SET DEFAULT VALUES** - prestaví hodnoty všetkých voliteľných hodnôt naspäť na pôvodné hodnoty.

V záložke **Thresholding** obr. 4b, sa nachádzajú všetky naimplementované prahovacie algoritmy, ktoré odstraňujú šum a delia dátá na objekt a pozadie. A následne môžu vstupovať ako počiatočná podmienka do segmentačnej metódy. Ide o dva globálne prahovacie algoritmy, ktoré závisia len na histograme a teda neberú na vstup žiadne parametre, iba pôvodné dátá. Tri lokálne adaptívne prahovacie algoritmy, ktoré berú na vstup veľkosť masky, veľkosť časového kroku a pôvodné dátá. A zvyšné dva sú vytvorené ako hybridné za použitia lokálnych adaptívnych prahovacích algoritmov, tiež pri nich na vstup okrem pôvodných dát ide aj veľkosť masky a veľkosť časového kroku. Aplikovaná prahovacia funkcia berie na vstup vždy pôvodné dátá, to znamená prvé 2D dátá uvedené v zozname.



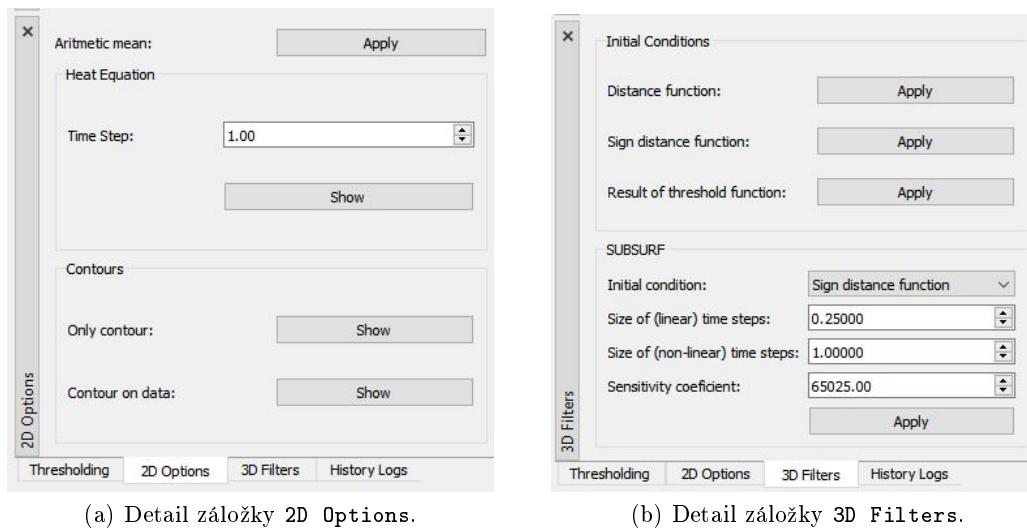
Obr. 4

Záložka **2D Options** obr. 5a je zobrazená len v prípade, že sú zvolené 2D dátá. V tejto záložke sa nachádzajú možnosti súvisiace len s 2D dátami. Dájú sa tam zobraziť kontúry na dátach, jeden krok rovnice vedenia tepla a priemer originálnych a vyprahovaných obrazových dát. Tieto zobrazenia slúžia pre užívateľa len informatívne a vždy sa aplikujú na aktuálne dátá, to znamená na dátá vyznačené v zozname.

3D Filters obr. 5b obsahuje možnosti pre samotný SUBSURF spolu s jeho parametrami. Taktiež sa tam nachádzajú počiatočné podmienky znamienkovej dištančnej a prahovacej funkcie, ktoré sa dajú zobraziť ešte pred vstupom do SUBSURFu. Počiatočné podmienky majú iba informatívny charakter. Pri segmentačnej metóde SUBSURF sa nachádza viacero voliteľných parametrov:

- Voľba počiatočnej podmienky, bud' ako znamienková funkcia vzdialenosť alebo vytvorená z výsledku prahovacej funkcie. Na vstup sa berú vždy aktuálne zvolené dátá zo zoznamu,

- Typ hranového detektoru, s ktorým sa bude SUBSURF počítať (obe možnosti popísané v sekcií 2.3.1). Tieto dátá treba najskôr zvoliť, inak sa na vstup budú brať aktuálne zvolené dátá zo zoznamu.
- Veľkosť lineálneho časového kroku σ , vstupujúceho ako parameter do rovnice vedenia tepla slúžiacej na predvyhladenie dát.
- Veľkosť nelineálneho časového kroku τ , vstupujúceho ako parameter do samotného SUBSURFu.
- Koeficient sensitivity, potrebný pri výpočte gradientov.

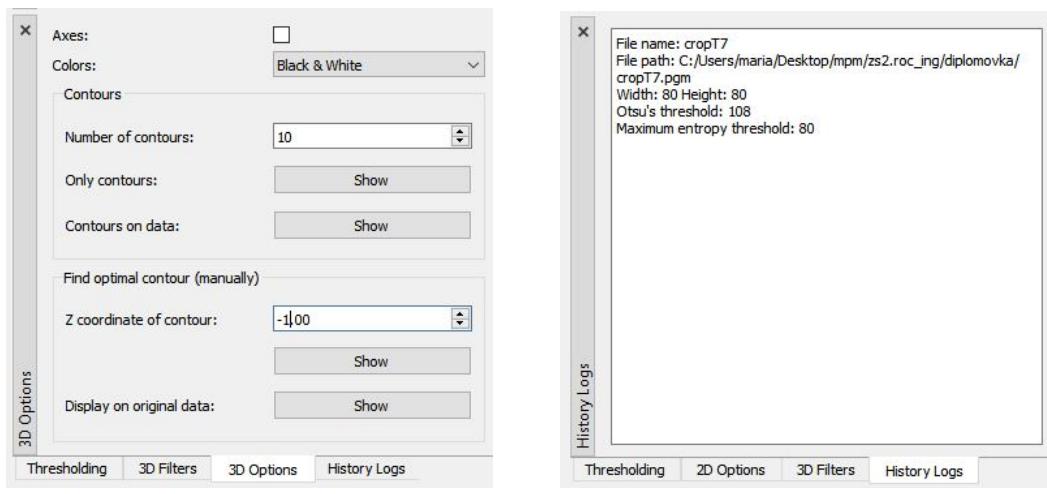


(a) Detail záložky 2D Options.

(b) Detail záložky 3D Filters.

Obr. 5

V 3D Options obr. 6a je zobrazený len v prípade, že sú zvolené 3D dátá. V tejto záložke sa nachádzajú možnosti súvisiace len s manipuláciou s 3D dátami. V týchto možnostiach sa dajú zapnúť osi, zmeniť farbu, vykresliť zvolený počet izočiar budť na dátach alebo bez nich a dá sa zvolať optimálna izočiara a následne ju je možné vykresliť na pôvodných dátach. Vykreslenie na pôvodných dátach by sa dalo považovať za konečný výsledok.



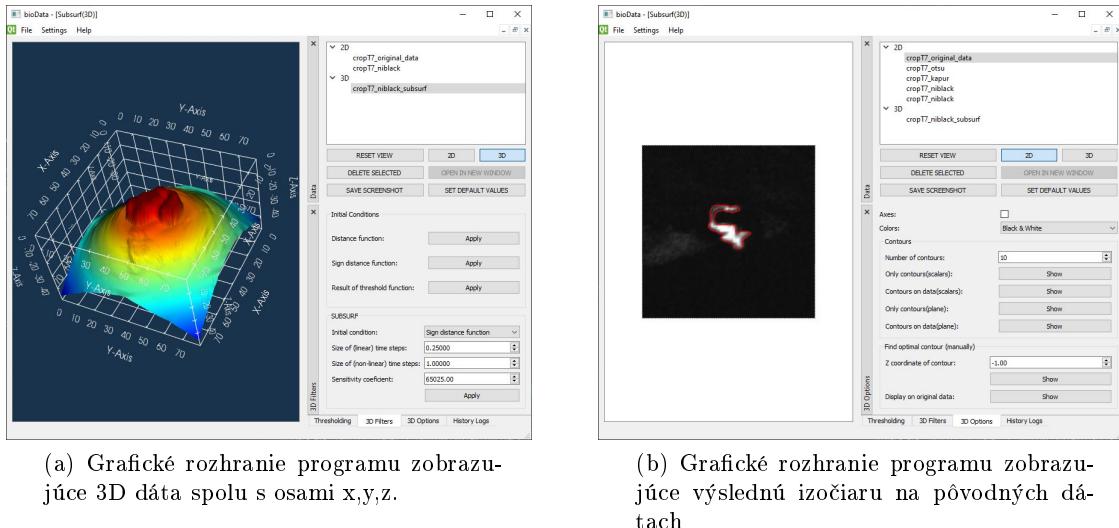
(a) Detail docku 3D Options.

(b) Detail docku History Logs.

Obr. 6

V záložke **History Logs** obr. 6b sa nachádza jednoduché textové pole, ktoré zaznamenáva väčšinu úkonov vykonaných v programe. Taktiež sa tam nachádzajú informácie o načítanom súbore, ako napríklad aj názov súboru, cesta k súboru atď... Tento výstup sa dá uložiť do jednoduchého textového súboru.

Na obr. 7a sú zobrazené dátá s makrofágom, na ktoré bol aplikovaný SUBSURF, za počiatocnú podmienku bola použitá dištančná funkcia.



Obr. 7

Na obr. 7b sú vidieť pôvodné dátá, na ktorých je zobrazená výsledná izočiara.

V prípade, že je v programe načítaných viac ako jeden súbor, tak sa v bočnom paneli zobrazí zláložka **Opened Windows** so zoznamom, v ktorom sa nachádzajú všetky súbory a pri zvolení niektorých dát zo zoznamu sa maximalizuje vykreslovacie podokno a aktualizujú sa informácie uložené v bočnom paneli.

Vykresľovacia plocha má niekoľko, vstavaných funkcií, ktoré sa dajú ovládať cez klávesové skratky. Ide napríklad o klávesy **W** a **S** prepínajúce zobrazenie dát medzi obrysom dát a zafarbeným modelom, klávesa **R** reštartuje pohľad na objekt na vykresľovanej ploche.

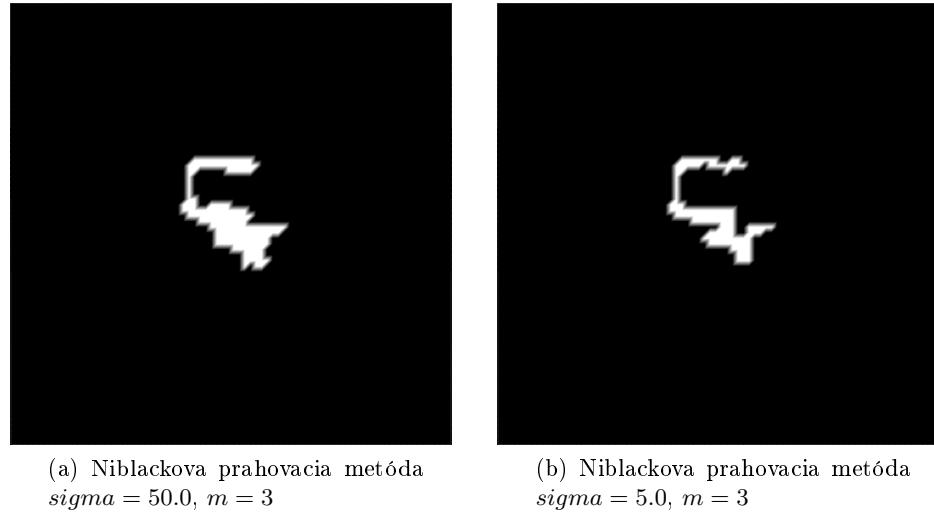
Zvyšné možnosti, v hornej lište v položke **File** sa ešte nachádza možnosť **Save**, ktorá ponúka buď exportovanie 2D aj 3D dát vo VTK formáte **.vtp** (typ súboru špecifický pre vtk-PolyData) alebo 2D dát vie uložiť ako ascii portable gray map dát **.pgm**. Ďalšou možnosťou je **Close Files**, kedy sa zatvoria všetky otvorené súbory. A možnosti **Close**, ktorá zatvorí celý program.

V hornej lište sa ešte nachádzajú položky **Settings** a **Help**. V **Settings** vie užívateľ skryť/zobraziť všetky položky nachádzajúce sa vpravo. A **Help** obsahuje informácie o programe a jednoduchú dokumentáciu.

4.5 Príklad použitia

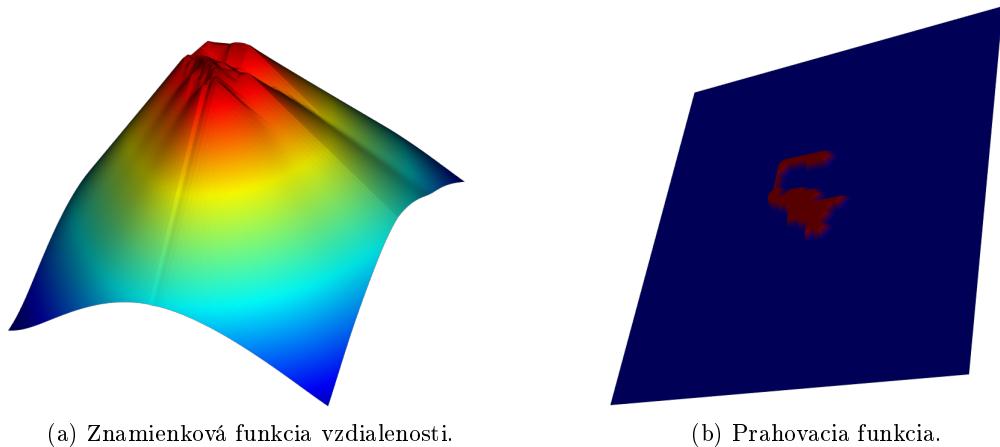
V tejto časti si ukážeme príklad na použitie implementovaného softvéru. Tento príklad bude ukážkou, ako sa dopracovať k čo najlepšiemu výsledku. Otvoríme súbor typu **.pgm**. Načíta sa užívateľské prostredie obr. 3. V záložke **Thresholding**, zvolíme niektorú z implementovaných prahovacích funkcií, v tomto príklade sme zvolili Niblackovu metódu. Na získanie čo najlepšieho výsledku použijeme dve rôzne prahovania. Prvé prahovanie bude s časovým krokom $\sigma = 50.0$ a veľkosťou masky $m = 3$, toto prahovanie pôjde na vstup do počiatocnej podmienky. Druhé prahovanie bolo zvolené s menším časovým krokom $\sigma = 5.0$ a rovnakou veľkosťou masky $m = 3$, toto prahovanie pôjde na vstup pre výpočet hranového

detektoru a slúži na vylepšenie vektorového poľa. Výsledoky oboch prahovaní je zobrazený na obr. 8.



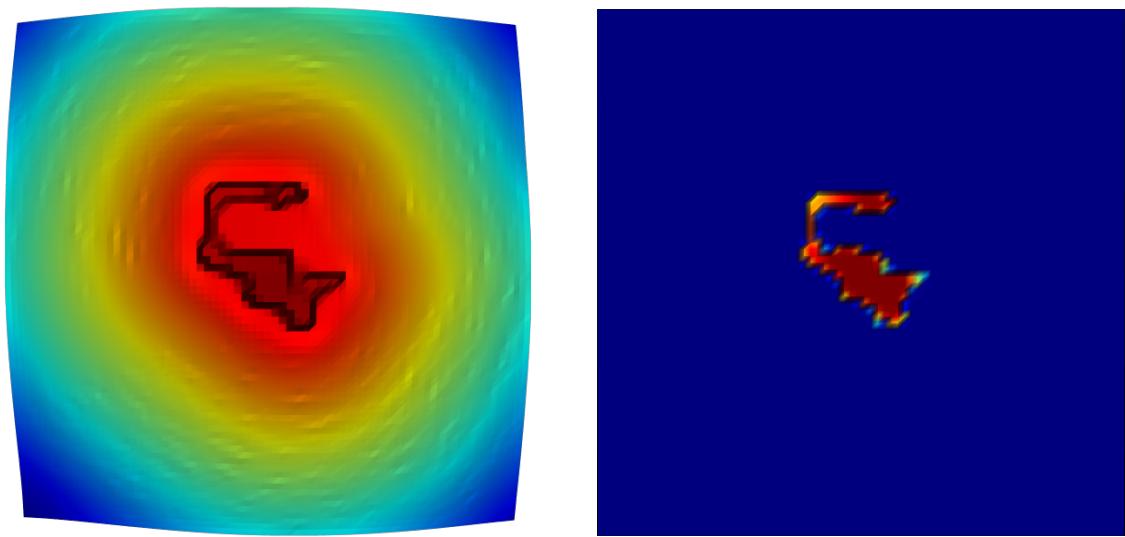
Obr. 8

Vyprahované dátá, boli pridané do zoznamu. Tieto dátá použijeme na vstup do segmentačnej metódy SUBSURF. Zvolíme záložku **3D Filters** detail možností je zobrazený na obr. 5b, kde sa dajú zobrazíť aj 2 funkcie, ktoré môžeme použiť ako počiatočnú podmienku do segmentačnej metódy, so vstupnými dátami zobrazenými na obr. 8. Prvou možnosťou počiatočnej podmienky je znamienková funkcia vzdialenosťi obr. 9a. Druhá možnosť je výsledok prahovacej funkcie obr. 9b, čo znamená, že takéto dátá majú nastavené súradnice z na -1 , ak prahovacia funkcia zaradila pixel do pozadia a 1, ak sa jedná o pixel makrofágu. Prahovacia funkcia idúca na vstup, musí byť v oboch prípadoch vyznačená v paneli **Data**.



Obr. 9

Typ počiatočnej podmienky, vstupujúcej do segmentačnej metódy SUBSURF zvolíme zo zoznamu. Následne nastavíme požadované parametre metódy: počet časových krokov t , veľkosť lineárneho časového kroku σ , veľkosť nelineárneho časového kroku τ , koeficient sensitivity k a typ hranového detektora. V ukážkovom príklade obr. 10 sú použité hodnoty: $t = 10$, $\sigma = 0.5$, $\tau = 1.0$, $k = 100$, hranový detektor 33, pre oba druhy počiatočných podmienok. Hranový detektor 33, sme zvolili pretože vracia lepšie výsledky.

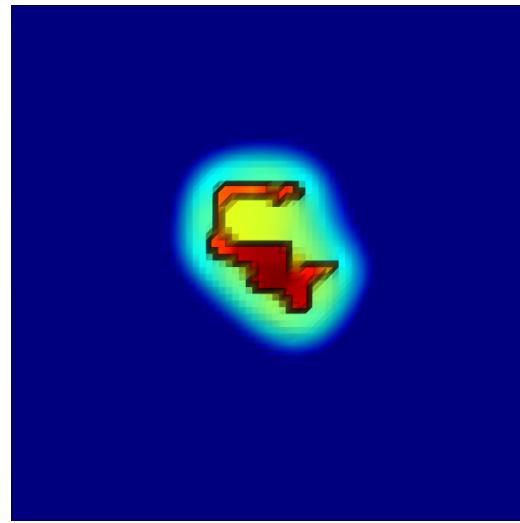


(a) Výsledok segmentačnej metódy pri použití znamienkovej funkcie vzdialenosť ako počiatočnej podmienky.

(b) Výsledok segmentačnej metódy pri použití prahovacej funkcie ako počiatočnej podmienky.

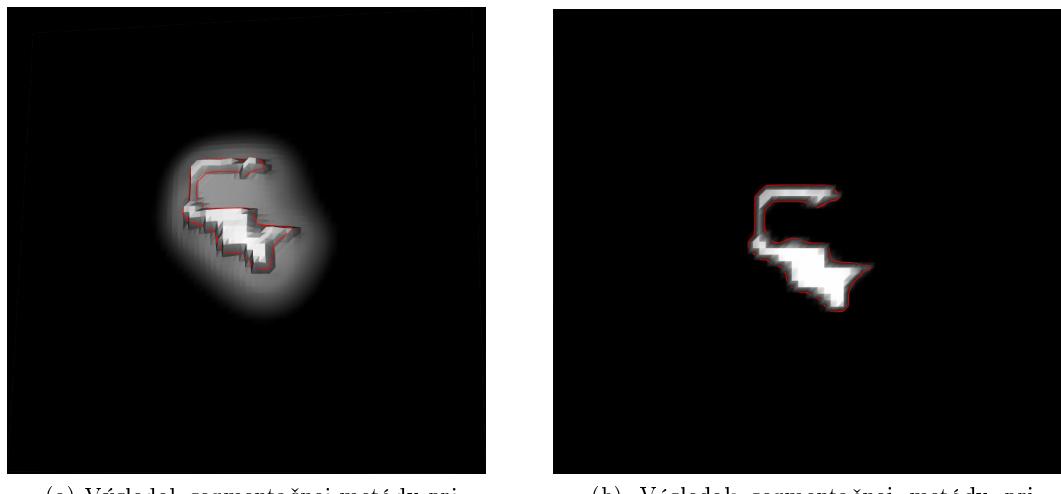
Obr. 10

Na obr. 10a je zobrazený výsledok s počiatočnou podmienkou znamienkovej funkcie vzdialenosť. Môžeme si všimnúť, že informácia o výsledku segmentácie nebude ovpluvnená časťou znamienkovej funkcie vzdialenosť, ktorá zostala na dátach po vysegmentovaní makrofágu. Preto môžeme výsledok segmentácie približne od hodnoty -8 orezať a všetky hodnoty sa nachádzajúce sa pod ňou budú mať hodnotu -8 . Možnosť na orezanie dát sa nachádza v záložke **3D Options** obr. 6 a výsledok je zobrazený na obr. 11.



Obr. 11: Orezaný výsledok segmentácie.

Teraz zvolíme optimálnu izočiaru, v prípade počiatočnej podmienky znamienkovej funkcie vzdialenosť zvolíme izočiaru s hodnotou $z = -1.5$ obr. 12a. V prípade počiatočnej podmienky prahovacej funkcie bude mať izočiara hodnotu $z = -0.5$ obr. 12b.

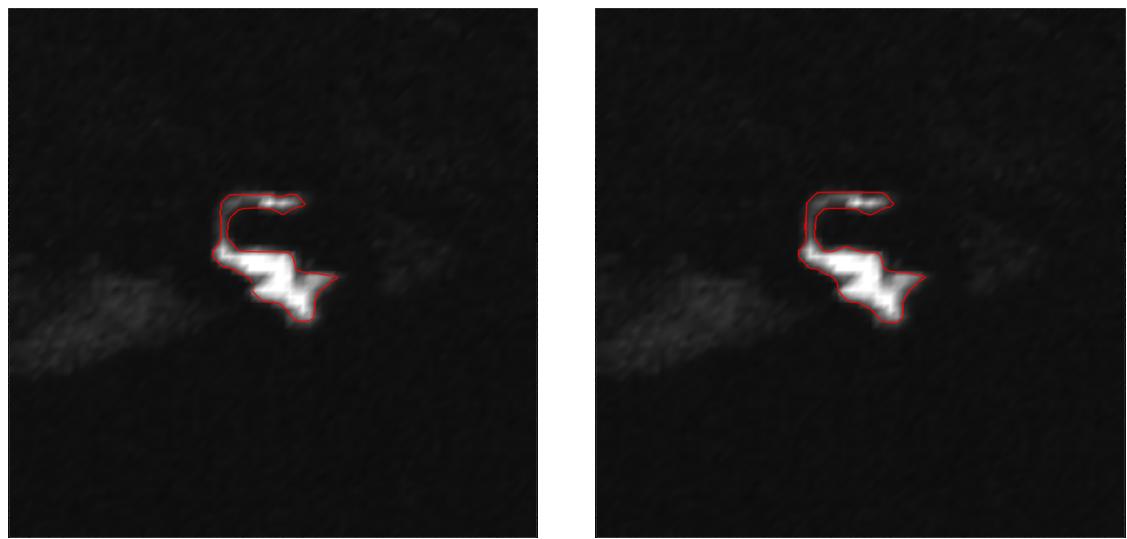


(a) Výsledok segmentačnej metódy pri použití znamienkovej funkcie vzdialenosť ako počiatočnej podmienky s izočiarou na hodnote $z = -1.5$.

(b) Výsledok segmentačnej metódy pri použití prahovacej funkcie ako počiatočnej podmienky s izočiarou na hodnote $z = -0.5$.

Obr. 12

Nakoniec optimálne zvolenú izočiaru vykreslíme obr. 13 spolu s pôvodnými dátami. Výsledné dáta vieme uložiť pomocou **SAVE SCREENSHOT** tlačidla vo formáte *.png*.



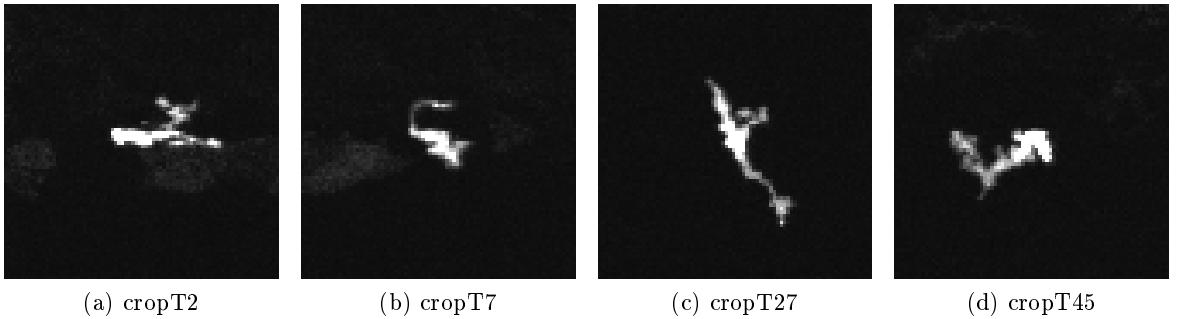
(a) Výsledná izočiara na pôvodných dátach pre počiatočnú podmienku znamienkovej funkcie vzdialenosť.

(b) Výsledná izočiara na pôvodných dátach pre počiatočnú podmienku prahovacej funkcie.

Obr. 13

5 Výsledky

Na ukážku výsledkov sme vybrali niekoľko dát s makrofágmi so zložitými tvarmi a šumom v pozadí. Vybrali sme nasledujúce dátá. Obr. 14a bol vybraný z dôvodu, že niektoré časti sú 'odtrhnuté' od zvyšku makrofágu a dátá obsahujú výrazný šum, viditeľný priamo za makrofágom. Dátá na Obr. 14b a Obr. 14c majú užšie časti, so slabšími intenzitami, ktoré je tiež potrebné zahrnúť ako časť makrofágu. Dátá 14d majú na ľavej časti výrazne slabšiu intenzitu.

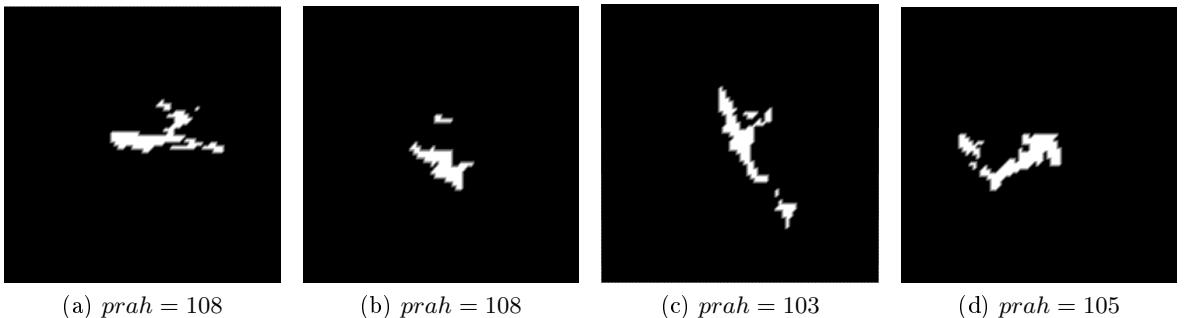


Obr. 14: Zvolené originálne dátá.

Na vstup do segmentačnej metódy sme implementovali dva druhy počiatočných podmienok, dva spôsoby výpočtu hranového detektora a niekoľko prahovacích metód popísaných v sekcií 2. Úlohou bolo, nájsť počiatočnú podmienku spolu s prahovacou metódou, ktoré by čo najpresnejšie popisovali pôvodný tvar makrofágu. V tejto sekcií si ukážeme najlepšie výsledky, ktoré sme dosiahli.

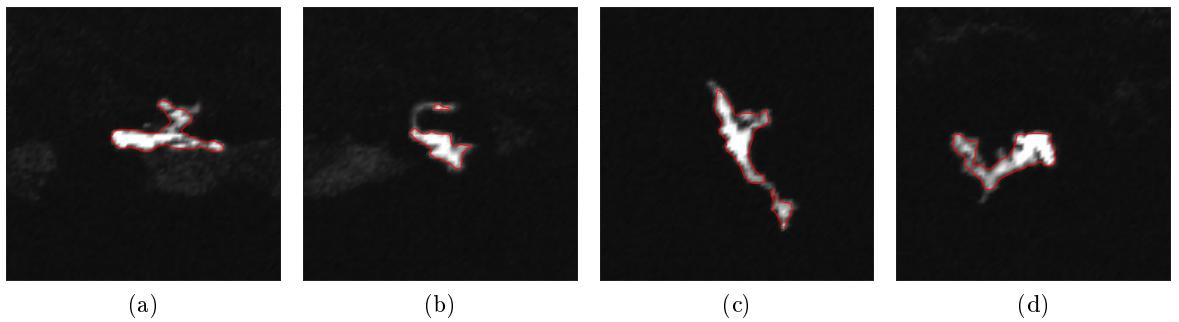
Parametre segmentačnej metódy SUBSURF boli zvolené nasledovne: $t = 15$, $\sigma = 0.5$, $\tau = 1.0$, $k = 500$, hranový detektor definovaný ako 31 a hodnotu výslednej izočiary sme volili podľa toho aký typ počiatočnej podmienky bol zvolený. Nami zvolená modifikácia hranového detektora je súčasťou pomalšia ale mala lepšie výsledky. Pri počiatočnej podmienke danej znamienkovou funkciou vzdialenosť sme izočiaru nastavili na hodnotu $z = -1.5$ a pri počiatočnej podmienke danej prahovacou funkciou bola izočiara nastavená na $z = -0.5$, keďže vo väčšine prípadoch boli tieto voľby najvhodnejšimi. Výsledky segmentácie sú vyznačené na pôvodných dátach červenou farbou.

Na obr. 15 je zobrazená globálna prahovacia funkcia popísaná v sekcií 2.1.1 používajúca Otsuho metódu. Dá sa všimnúť, že táto prahovacia metóda nezachytila celý tvar makrofágu, hlavne ak išlo o úzku časť s menšou intenzitou.



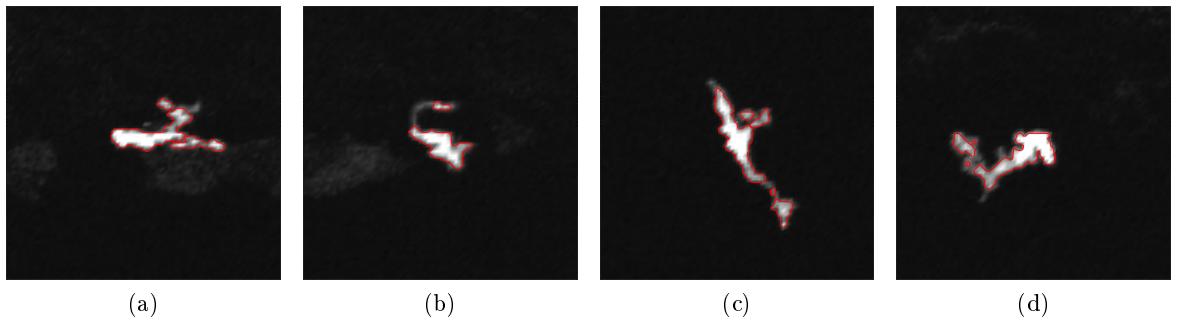
Obr. 15: Otsuho metóda

Izočiara na obr. 16 vyznačujúca výsledok segmentácie s počiatočnou podmienkou znamienkovej funkcie vzdialenosť. Môžeme si všimnúť, že niektoré časti makrofágu odtrhnuté aplikáciou prahovacej metódy, sa segmentáciou späť spojili.



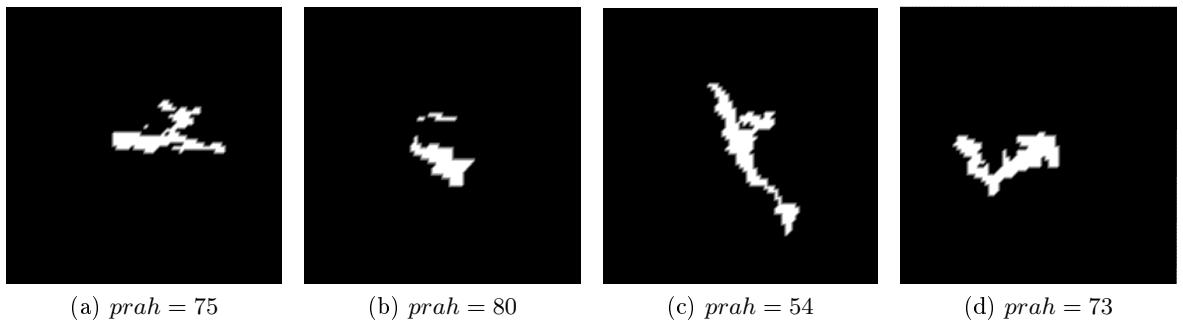
Obr. 16: Výsledok segmentácie. Použitá Otsuho metóda a znamienková funkcia vzdialenosť.

Na obr. 17 vidíme výsledok segmentačnej metódy s počiatočnou podmienkou vytvorennej z prahovacej metódy. Tato počiatočná podmienka nevylepšila výsledok, výsledky sú takmer totožné s s výsledkom prahovacej metódy.



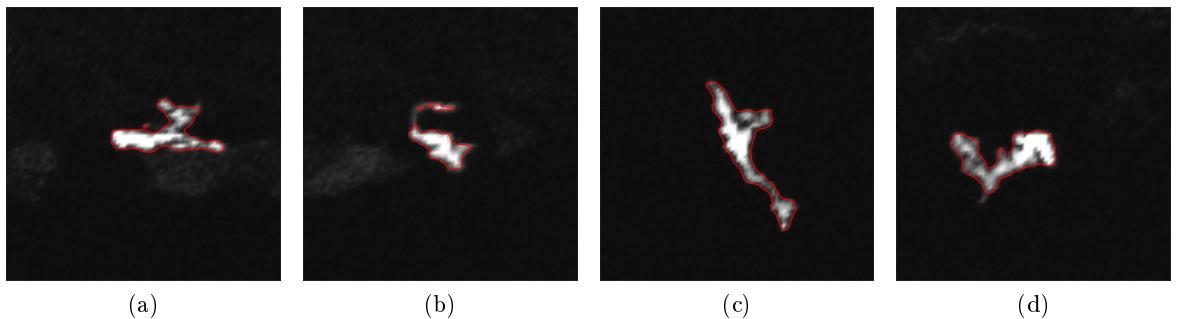
Obr. 17: Výsledok segmentácie. Použitá Otsuho metóda a prahovacia funkcia.

Ďalšou globálnou prahovacou metódou popísanou v sekcií 2.1.2, je metóda získavajúca prah pomocou maximálnej entropie zobrazená na obr. 18. Môžeme si všimnúť, že prah tejto metódy bol určený o niečo vhodnejšie(nižšie) ako predošlá metóda, napríklad v prípade 18c. Avšak pri tomto type prahovacej metódy sa môže stať, že makrofág bude obsahovať aj časti, ktoré pôvodne nemal. Z výsledku prahovania obr. 18c si môžeme všimnúť, že makrofág má pomerne jednoduchý tvar a prahovacia metóda zachytila jeho tvar takmer dokonale.



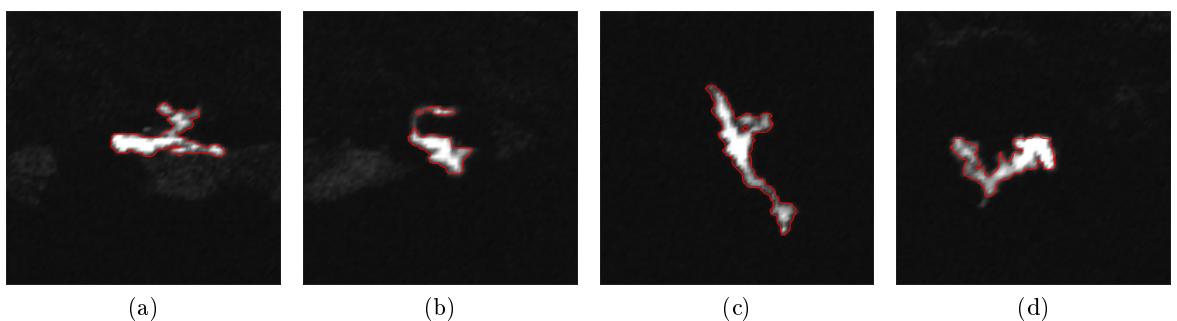
Obr. 18: Prahovanie pomocou maximálnej entropie.

Výsledky segmentačnej metódy s počiatočnou podmienkou znamienkovej funkcie vzdialenosťi obr. 19.



Obr. 19: Výsledok segmentácie. Použité prahovanie pomocou maximálnej entropie a znamienková funkcia vzdialenosťi.

Výsledky segmentačnej metódy s počiatočnou podmienkou prahovacej funkcie obr. 20.

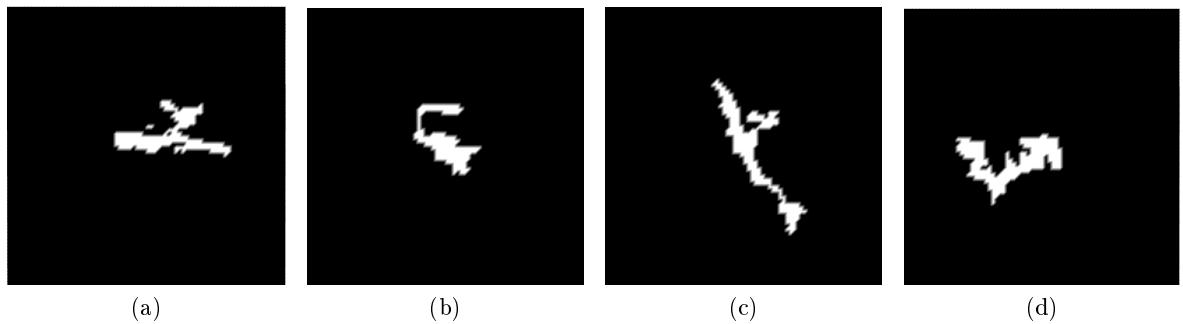


Obr. 20: Výsledok segmentácie. Použité prahovanie pomocou maximálnej entropie a prahovacia funkcia.

Lepšie výsledky získane z aplikácií globálnych prahovacích metód na dátu získame pomocou nájdením prahu cez maximálnu entropiu a počiatočnej podmienky znamienkovej funkcie vzdialenosťi.

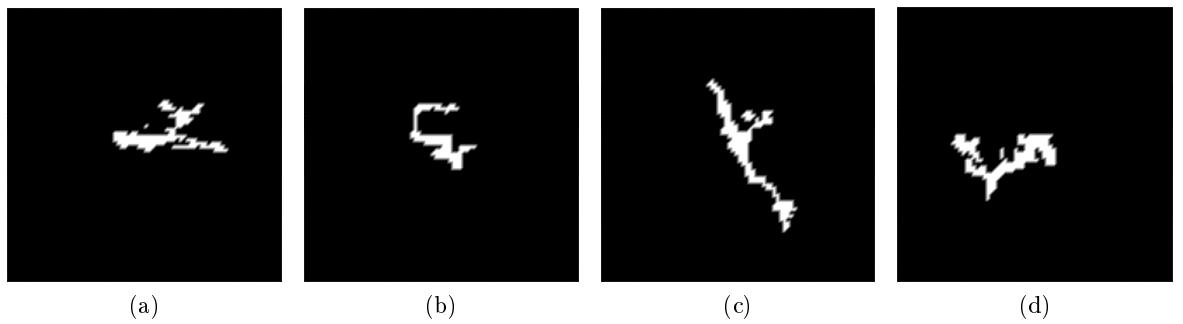
V prípade lokálnych adaptívnych prahovacích metód je prah vyhodnotený pre každý pixel osobitne na nejakom fixne danom okolí, teda sme predpokladali, že výsledky týchto metód budú viac vyhovovať pri segmentácii makrofágov. Avšak tieto metódy závisia minimálne od veľkosti masky m , na ktorej sa počíta hodnota prahu pre zvolený pixel. A v niektorých prípadoch sa približne počíta priemer intenzít a smerodajná odchýlka pomocou jedného kroku rovnice vedenia, teda ďalším prametrom je veľkosť časového kroku σ .

Prvou lokálnou adaptívnu metódou bola Niblackovej prahovacia metóda popísaná v sekcií 2.2.2. S veľkosťou časového kroku $\sigma = 50.0$ a veľkosťou masky $m = 3$, výsledky sú zobrazené na obr. 21. Výsledok tohto prahovania použijeme ako vstupné dátu do počiatočnej podmienky.



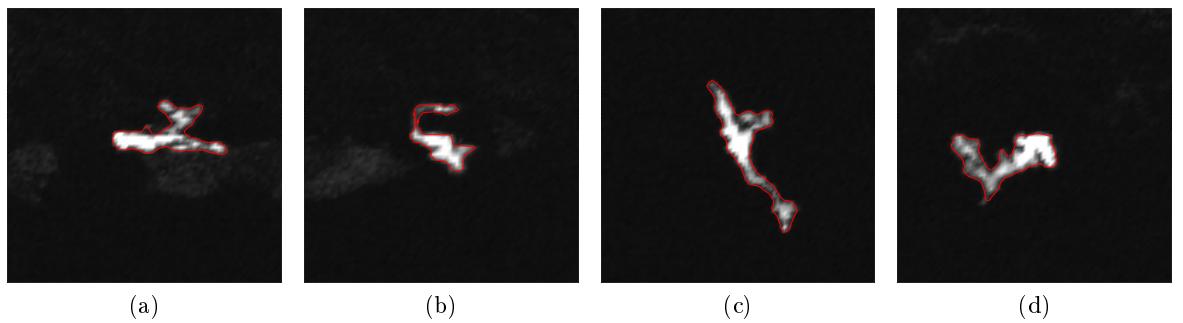
Obr. 21: Niblackova segmentačná metóda, pre $\sigma = 50.0$ a $m = 3$.

Na pôvodné dátá aplikujeme Niblackovu prahovaciu metódu ešte raz s menším časovým krokom ako v predošлом prípade $\sigma = 5.0$ obr. 22. Z výsledku tohto prahovania sa bude počítať hranový detektor a slúži na vylepšenie vektorového poľa.



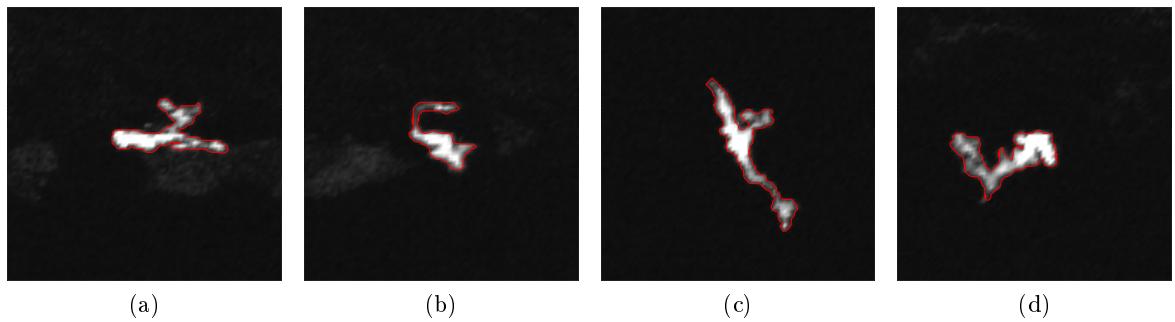
Obr. 22: Niblackova segmentačná metóda, pre $\sigma = 5.0$ a $m = 3$.

V oboch prípadoch si môžeme všimnúť, že výsledky prahovacích metód výrazne lepšie zachytávajú tvar makrofágov aj na miestach s nižšou intenzitou naroziel od globálnych prahovacích metód. Výsledky segmentačnej metódy s počiatočnou podmienkou znamienkovej funkcie vzdialenosťi obr. 23.



Obr. 23: Výsledok segmentácie. Použitá Niblacková metóda a znamienková funkcia vzdialenosťi.

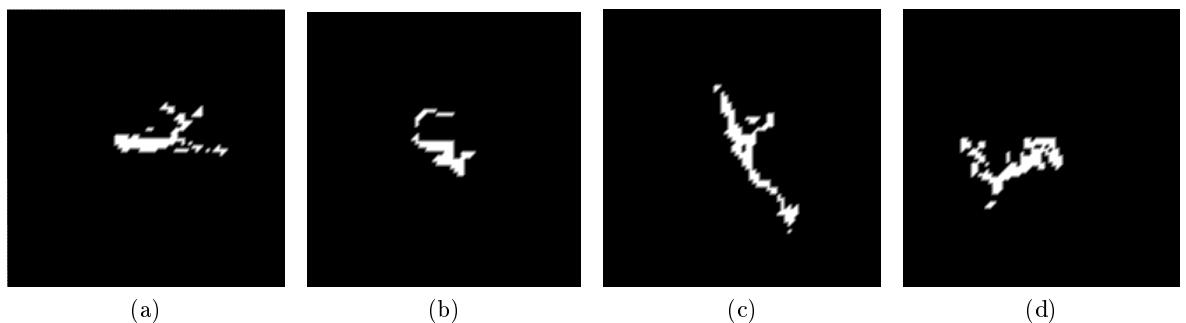
Výsledky segmentačnej metódy s počiatočnou podmienkou prahovacej funkcie obr. 24.



Obr. 24: Výsledok segmentácie. Použitá Niblacková metóda a prahovacia funkcia.

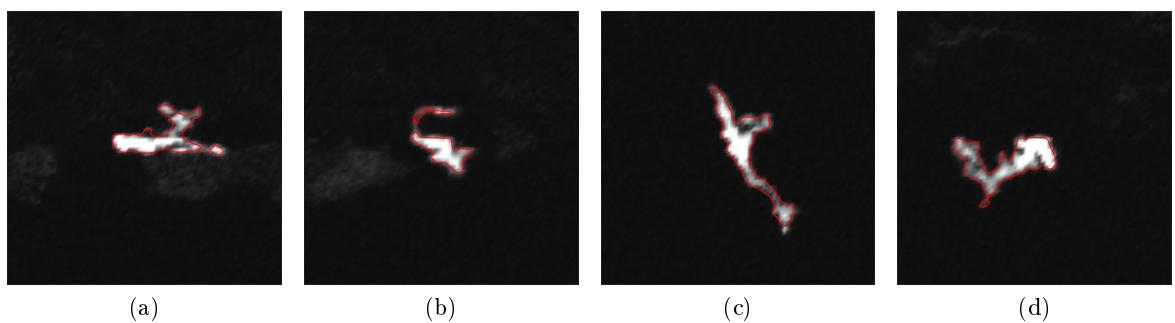
Z výsledkov segmentácie vidieť, že táto metóda veľmi dobre popisuje tvar makrofágov, aj keď obsahujú slabšiu intenzitu, o šum priamo za objektom alebo majú zložitý tvar.

Bernsenova prahovacia funkcia popísaná v sekcií 2.2.1, pre veľkosť masky $m = 2$, je zoobrazená na 25.



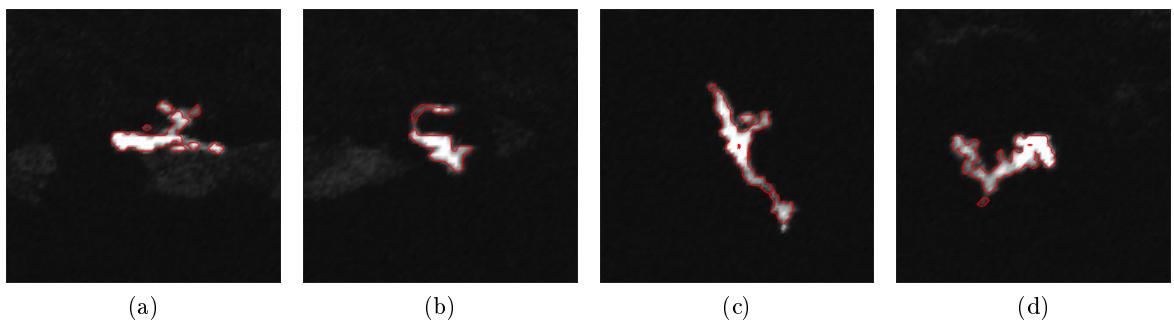
Obr. 25: Bernsenova metóda.

Výsledky segmentačnej metódy s počiatočnou podmienkou znamienkovej funkcie vzdialenosťi obr. 26.



Obr. 26: Výsledok segmentácie. Použitá Bernsenova metóda a znamienková funkcia vzdialenosťi.

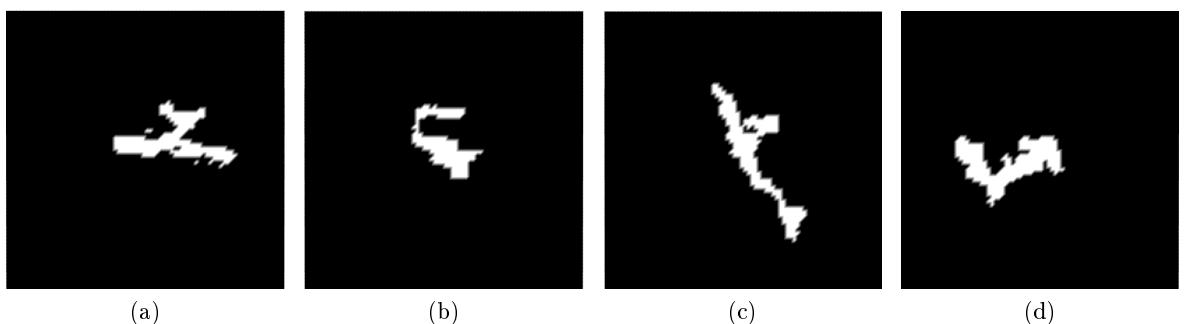
Výsledky segmentačnej metódy s počiatočnou podmienkou prahovacej funkcie obr. 27.



Obr. 27: Výsledok segmentácie. Použitá Bernsenova metóda a prahovacia funkcia.

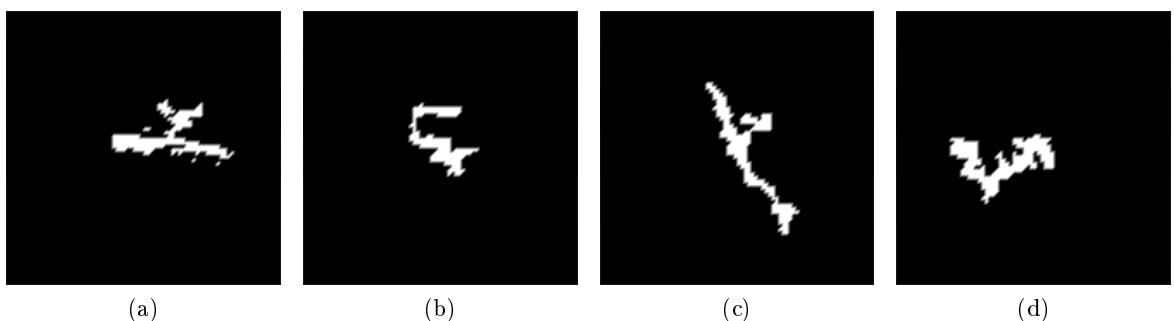
Z výsledkov vidieť, že táto prahovacia metóda nezachytí užšie časti s nižšiou intenzitou hoci niektoré časti boli doplnené po aplikácii segmentačnej metódy s počiatočnou podmienkou znamienkovej funkcie vzdialenosťi.

Sauvolova prahovacia metóda bola upravená z Niblackovej a používa sa primárne na prahovanie textových obrazových dát ale keďže makrofágy sú malé a majú nepravidelné tvary, tak sme predpokladali, že aj v našom prípade by mohla dávať uspokojuivé výsledky. Na obr. 29 s parametrami $\sigma = 50.0$ a $m = 3$. Výsledok tohto prahovania použijeme ako vstupné dátá do počiatočnej podmienky.



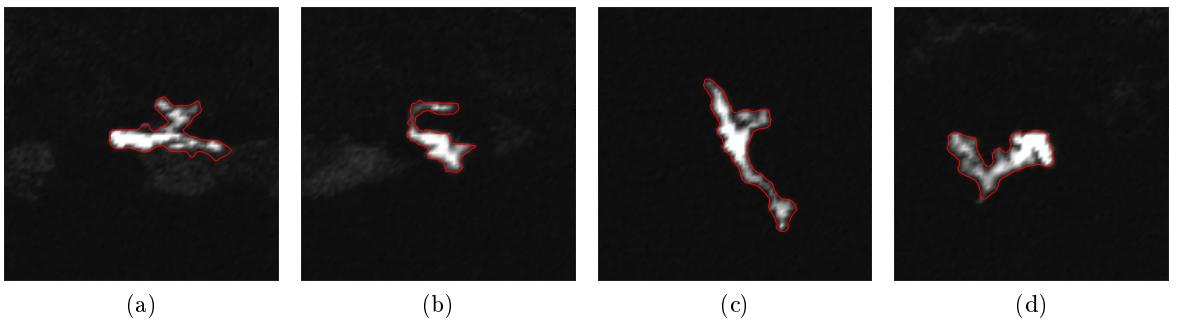
Obr. 28: Sauvolova metóda, pre $\sigma = 50.0$, $m = 3$.

Druhé prahovanie pomocou tejto metódy s parametrami $\sigma = 5.0$, $m = 3$, ktoré použijeme v segmentačnej metóde na výpočet hranového detektora.



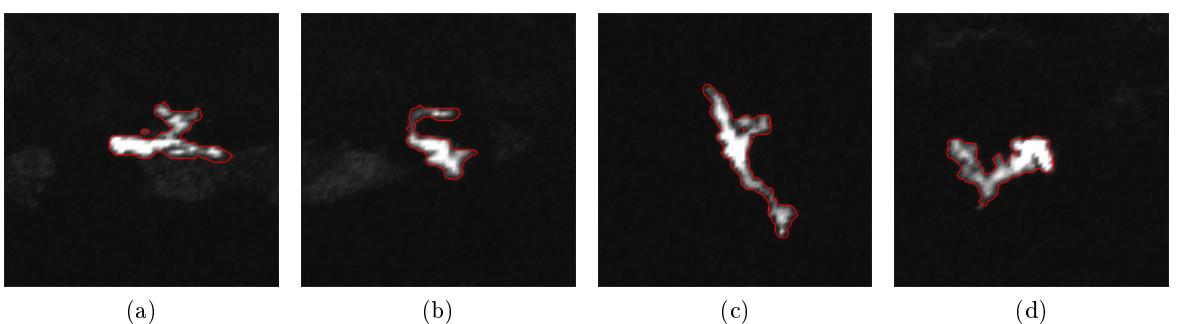
Obr. 29: Sauvolova metóda, pre $\sigma = 5.0$ a $m = 3$.

Výsledky segmentačnej metódy s počiatočnou podmienkou znamienkovej funkcie vzdialenosťi obr. 30.



Obr. 30: Výsledok segmentácie. Použitá Sauvolova metóda a znamienková funkcia vzdialosti.

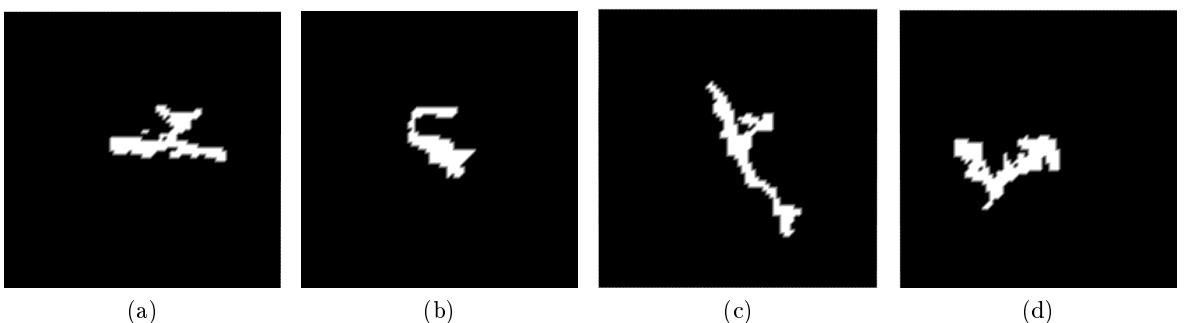
Výsledky segmentačnej metódy s počiatočnou podmienkou prahovacej funkcie obr. 31.



Obr. 31: Výsledok segmentácie. Použitá Sauvolova metóda a prahovacia funkcia.

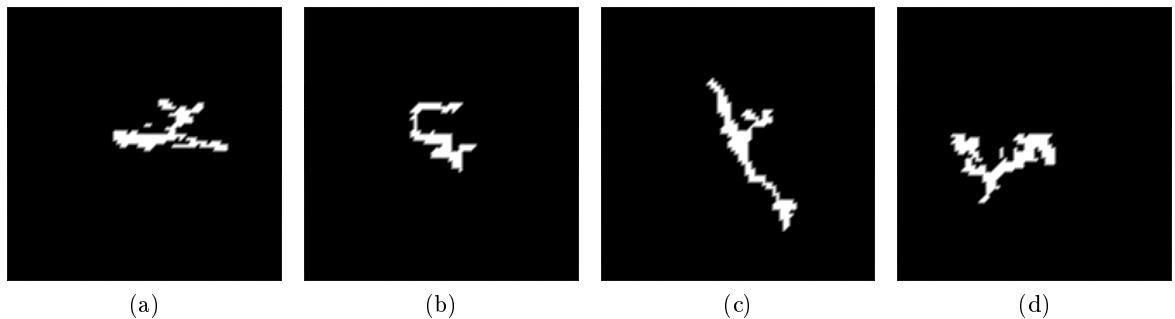
Na výsledkoch tejto segmentácií konkrétnie na obr. 30a a obr. 31a vidieť, že hoci sa prahovacia metóda nikde neroztrhla ale zachytila aj časť šumu nachádzajúcej sa na dátach.

Poslednými dvomi lokálnymi adaptívnymi metódami sú hybridné metódy používajúce princípy z predchádzajúcich metód. Prvá hybridná metóda vznikla kombináciou Bernsenovej a Nicblackovej metódy na obr. 32 sú zobrazené výsledky s parametrami $\sigma = 50.0$ a $m = 3$. Výsledok tejto metódy použijeme ako vstup do počitočnej podmienky.



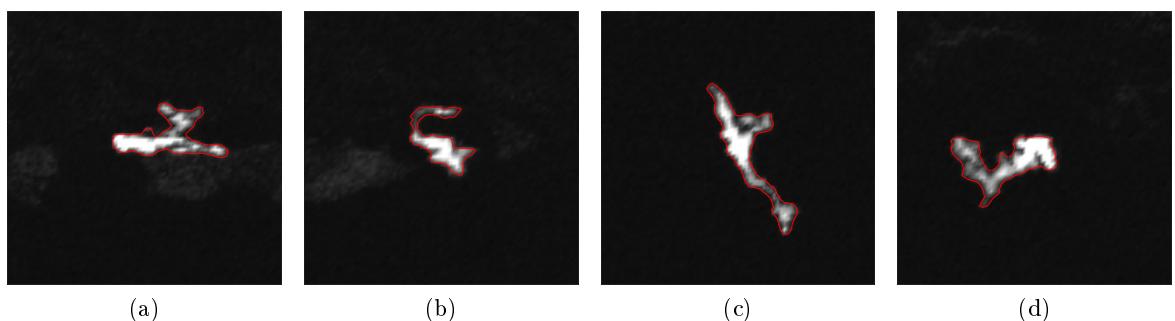
Obr. 32: Hybridná Bernsenova a Niblackova metóda, pre $\sigma = 50.0$, $m = 3$.

Následne spravíme ešte jedno prahovanie obr. 33 s parametrami $\sigma = 5.0$, $m = 3$, bude použité pri výpočte hranového detektora a vylepší vektorové pole pri samotnej segmentácii.



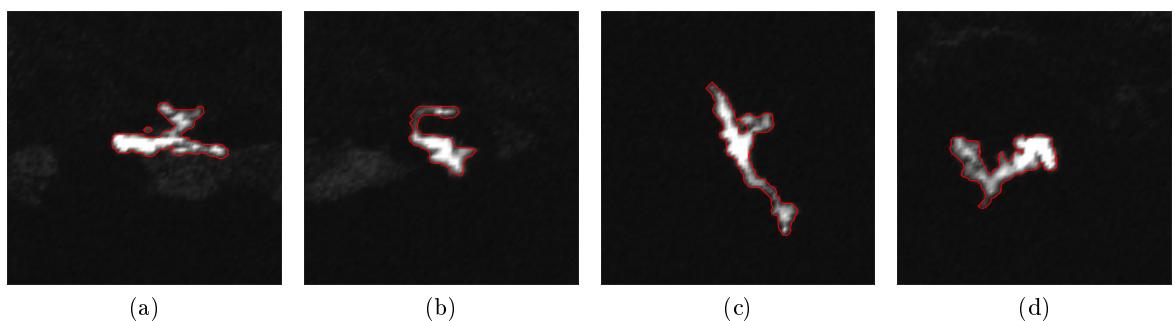
Obr. 33: Hybridná Bernsenova a Niblackova metóda, pre $\sigma = 5.0$, $m = 3$.

Výsledky segmentačnej metódy s počiatočnou podmienkou znamienkovej funkcie vzdialenosťi obr. 34.



Obr. 34: Výsledok segmentácie. Použitá prvá hybridná prahovacia metóda a znamienková funkcia vzdialenosťi.

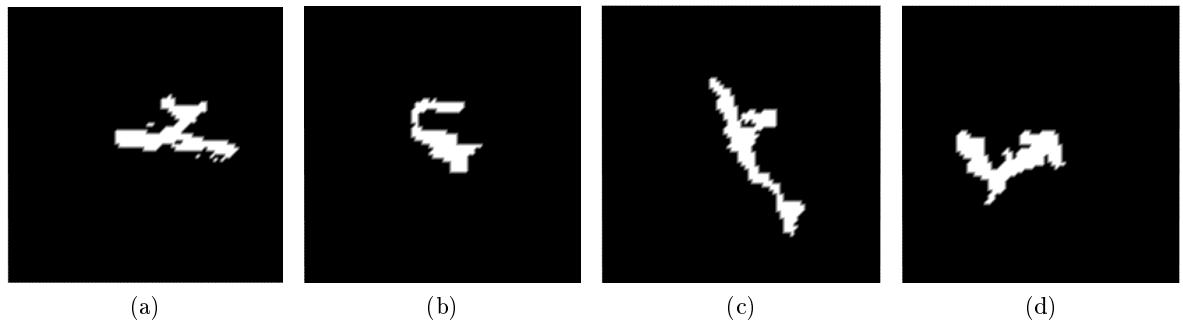
Výsledky segmentačnej metódy s počiatočnou podmienkou prahovacej funkcie obr. 35.



Obr. 35: Výsledok segmentácie. Použitá prvá hybridná metóda a prahovacia funkcia.

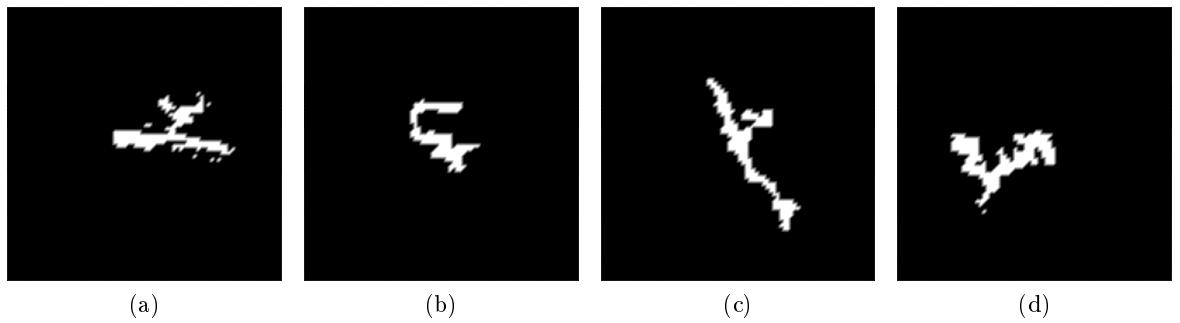
Výsledky tejto segmentácie veľmi dobre popisujú skutočný tvar makrofágov.

Druhá hybridná metóda vznikla kombináciou Bernsenovej a Sauvоловej metódy na obr. 36 sú zobrazené výsledky s parametrami $\sigma = 50.0$ a $m = 3$. Výsledok tohto prahovania sme použili ako vstup na výpočet počiatočnej podmienky.



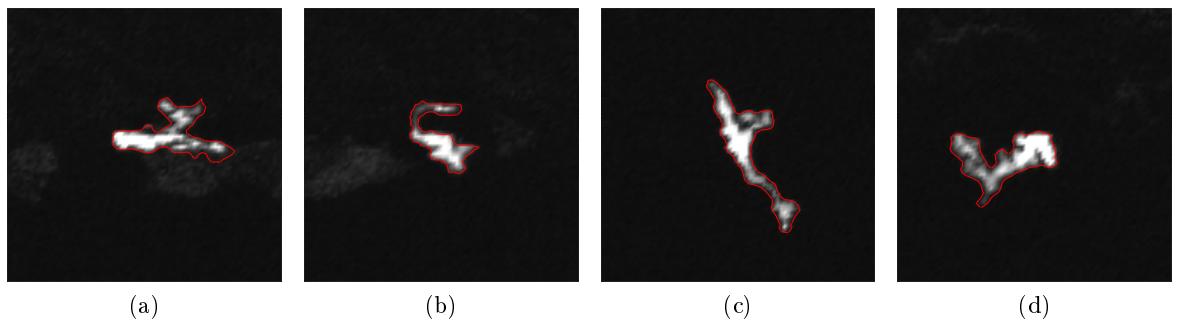
Obr. 36: Hybridná Bernsenova a Sauvolova metóda, pre $\sigma = 5.0$, $m = 3$.

Tu sme opäť použili dve rôzne prahovania, druhé malo parametre $\sigma = 5.0$, $m = 3$ vstu-povalo na výpočet hranového detektora. Výsledok tohto prahovania je zobrazený na obr. 37



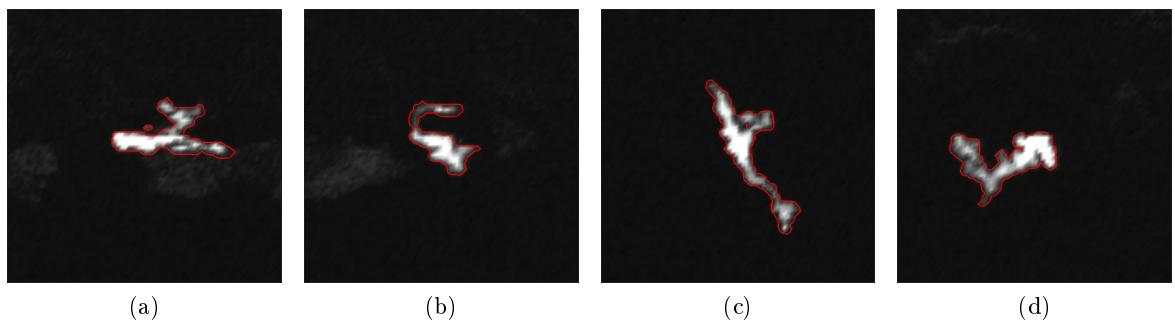
Obr. 37: Hybridná Bernsenova a Sauvolova metóda, pre $\sigma = 5.0$ a $m = 3$.

Výsledky segmentačnej metódy s počiatočnou podmienkou znamienkovej funkcie vzdialenosťi obr. 38.



Obr. 38: Výsledok segmentácie. Použitá druhá hybridná prahovacia metóda a znamienková funkcia vzdialenosťi.

Výsledky segmentačnej metódy s počiatočnou podmienkou prahovacej funkcie obr. 39.

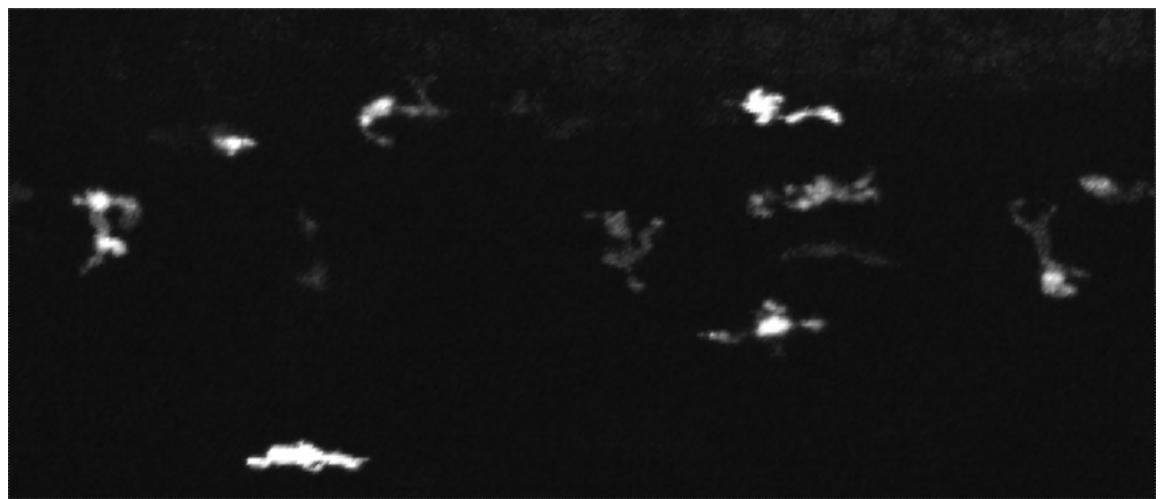


Obr. 39: Výsledok segmentácie. Použitá druhá hybridná metóda a prahovacia funkcia.

Výsledky tejto segmentácie sú podobné Sauvolovej metóde a nachádza sa okolo dát zo statkový šum.

Zo zobrazených výsledkov si môžeme všimnúť že výrazne lepšie výsledky získame pri použití niektorých z lokálnych adaptívnych prahovacích metód, ale napr. pri Sauvolovej prahovacnej funkcií je zachytený aj šum patriaci pozadiu. Najlepšie výsledky sme dostali použitím počiatočnej podmienky z hybridnej Bernsenovej a Niblackovej prahovej metódy ale aj Niblackova prahovacia metóda mala veľmi dobré výsledky.

Hoci sme primárne pracovali s dátami o veľkosti 80×80 , tak naimplementovaný program sa dá použiť aj pre všeobecnejšie dátá. Z pôvodných obrazových dát získaných mikroskopom, sme spravili výsek o veľkosti 350×150 obr. 40.



Obr. 40: Výsek viacerých makrofágov z pôvodných dát veľkosti 350×150 .

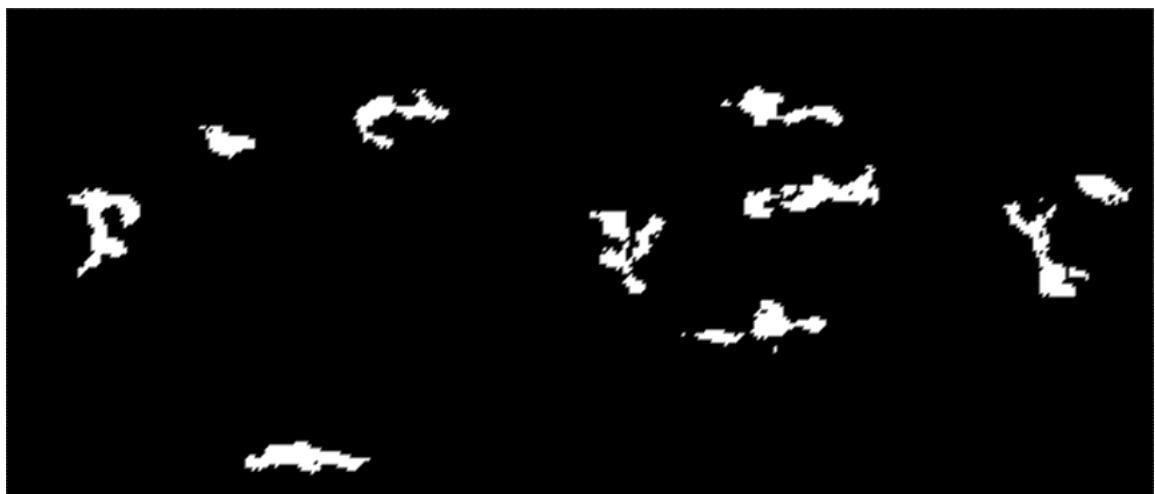
Na dátá na obr. 40 sme aplikovali segmentačnú metódu s najlepšiu globálnu a najlepšiu lokálnu adaptívnu prahovaciu metódu aby sme videli, či tieto metódy fungujú aj pre všeobecné prípady a vyhodnotili výsledky.

Z globálnych prahovacích metód sme zvolili, prahovanie pomocou maximálnej entropie obr. 41. Na výsledkoch z tohto prahovania si môžeme všimnúť, že makrofágy nemajú zachovaný tvar a niektoré, so slabšou intenzitou neboli vôbec zachytené.



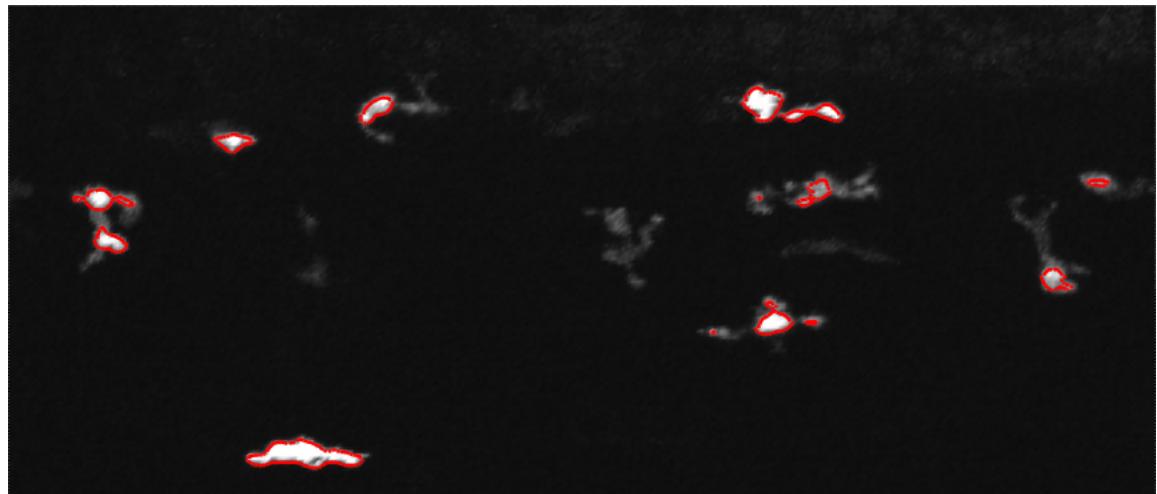
Obr. 41: Výsledok prahovania pomocou maximálnej entropie.

Ako lokálnu adaptívnu metódu sme zvolili hybridnú Bernsenovu a Niblackovu metódu s parametrami $\sigma = 50.0$, $m = 5$ na vstup do počiatočnej podmienky (obr. 44) a $\sigma = 5.0$, $m = 5$ ako vstup pri výpočte hranového detektora. Z výsledku vidieť že zachytené boli takmer všetky makrofágy až na veľmi malé so slabou intenzitou a táto metóda aj veľmi dobre zachytáva aj ich tvary.

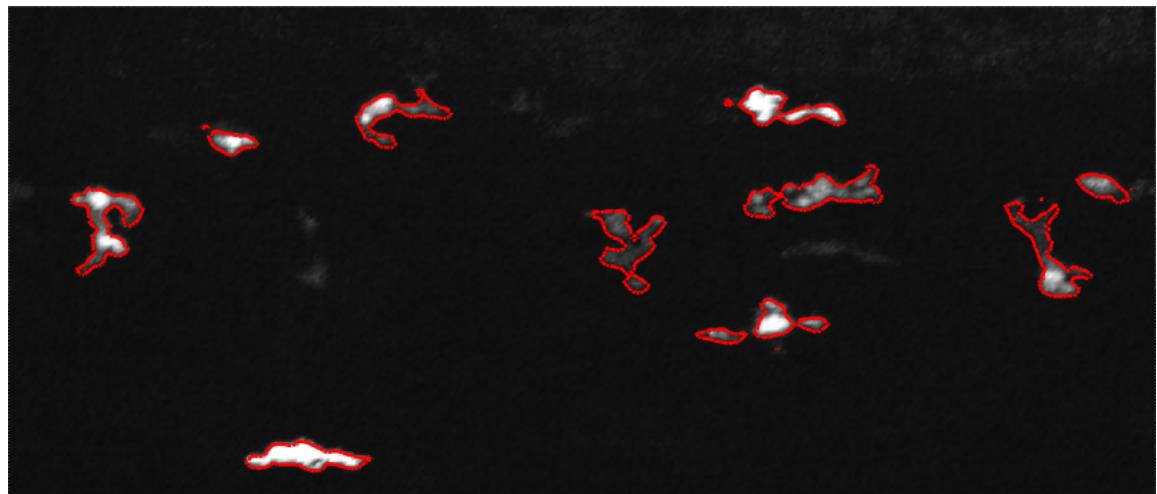


Obr. 42: Výsledok prahovania hybridnej Bernsenovej a Niblackovej metódy, pre $\sigma = 50.0$.

Pri samotnej segmentačnej metóde SUBSURF sme použili parametre: $t = 10$, $\sigma = 0.5$, $\tau = 1.0$, $k = 100$, hranový detektor 33 a počiatočnú podmienku znamienkovej funkcie vzdialenosťi.



Obr. 43: Výsledok segmentácie. Použité prahovanie pomocou maximálnej entropie a znamienková funkcia vzdialenosťi.



Obr. 44: Výsledok segmentácie. Použitá hybridná Bernsenova a Niblackova metóda a znamienková funkcia vzdialenosťi.

Z výsledkov segmentácie jednoznačne vidieť, že lokálna adaptívna metóda obr. 44 je vhodnejšia pri segmentácii makrofágov. Takmer presne zachytáva ich tvar aj na miestach s nižšou intenzitou. V prípade použitia prahovania pomocou maximálnej entropie si môžeme všimnúť, že sú zachytené len makrofágy alebo časti makrofágov len s vyššou intenzitou.

6 Záver

Cieľom tejto práce, bolo vytvoriť užívateľské prostredie s viacerými implementovanými segmentačnými metódami slúžiacimi na skúmanie tvarov a správania biologických dát - makrofágov. Pri implementácii bol použitý objektovo orientovaný jazyk C++, spolu s Qt knižnicami pomocou, ktorých bolo vytvorené užívateľské prostredie a VTK knižnicami ktoré zabezpečili zobrazenie a manipuláciu s dátami.

Implementovaných bolo niekoľko prahovacích metód, ktoré slúžia na zachytenie tvarov makrofágov a odstránenie šumu. K najlepším výsledkom, sme dospeli pri použití lokálnych adaptívnych prahovacích metód a to buď pri Niblackovej alebo hybridnej Bernsenovej a Niblackovej prahovacej metóde. Tieto prahovacie metódy zachytili tvar makrofágov veľmi dobre v prípadoch že sa na skúmaných dátach nachádzal výrazný šum, mali komplikovaný tvar ale aj v prípadoch keď sa vrámci jedného makrofágu výrazne menila intenzita. Tieto výsledky by sme mohli považovať za vylepšenie metód použitých v práci [3]. Tieto prahovacie metódy slúžili pri výpočte hranového detektou alebo mohli slúžiť ako počiatočná podmienka segmentačnej metóty SUBSURF.

Do budúcnosti by sa mohla funkcionálita naimplementovaného softvéru viac zo optimalizať. V prípade, ak by sa mal softvér využívať aj na dáta väčších rozmerov, tak by ho bolo kvôli výpočtovej náročnosti segmentačnej metódy vhodné sparalelniť napríklad použitím `openMP`. Tiež by sa dali pridať aj iné prahovacie metódy.

Literatúra

- [1] **VTK dokumentácia:** <https://vtk.org/documentation/>,
- [2] **Qt dokumentácia:** <https://doc.qt.io/>,
- [3] Seol Ah Park, Tamara Sipka, Zuzana Krivá, Martin Ambroz, Michal Kollar, Balazs Kosa, Mai Nguyen-Chi, Georges Lutfalla, Karol Mikula, **Macrophage image segmentation by Thresholding and subjective surface Method:** https://www.researchgate.net/publication/337755994_Macrophage_image_segmentation_by_Thresholding_and_subjective_surface_Method,
- [4] Zuzana Krivá, Karol Mikula, Oľga Stašová, **Spracovanie obrazu vybrané kapitoly z prednášok**,
- [5] Wilhelm Burger, **Principles of Digital Image Processing: Core Algorithms**,
- [6] Anna Korzynska, Lukasz Roszkowiak, Carlos Lopez, Ramon Bosch, Lukasz Witkowski1, Marylene Lejeune **Validation of various adaptive threshold methods of segmentation applied to follicularlymphoma digital images stained with 3,3'-Diaminobenzidine&Haematoxylin:** <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3656801/>,
- [7] Nobuyuki Otsu, **A Threshold Selection Method from Gray-Level Histograms**,
- [8] J.N.Kapur, P.K.Sahoo, A.K.C.Wong **A new method for gray-level picture thresholding using the entropy of the histogram**, [https://doi.org/10.1016/0734-189X\(85\)90125-2](https://doi.org/10.1016/0734-189X(85)90125-2)