

ŽILINSKÁ UNIVERZITA V ŽILINE

Fakulta riadenia a informatiky

DIPLOMOVÁ PRÁCA

Študijný program: Informačné systémy

Bc. Michal Kvet

**Transformácia dát
z magnetickej rezonancie hlavy
pre 3D vizualizáciu**

Vedúci: prof. Ing. Karol Matiaško, PhD.
Reg.č.: 123/2011 Máj 2012

Žilina

Abstrakt:

KVET, Michal: *Transformácia dát z magnetickej rezonancie hlavy pre 3D vizualizáciu* [diplomová práca] - Žilinská univerzita v Žiline. Fakulta riadenia a informatiky; Katedra informatiky. - Vedúci: prof. Ing. Karol Matiaško, PhD. – Stupeň odbornej kvalifikácie: Inžinier v študijnom odbore Aplikovaná informatika. Žilina: FRI ŽU v Žiline, 2012.

Výskum v oblasti medicíny prináša nové metódy diagnostiky a liečby ochorení. Rakovina v akejkoľvek forme patrí k závažným ochoreniam a často končí smrťou. Cieľom projektu „Detekcia nádoru mozgu“ je detegovať, lokalizovať anomálie mozgu a sledovať vývoj v čase. 26 základných markerov a neobmedzený počet markerov definovaných v aplikácii umožňuje detegovať široké spektrum rakovinotvorných buniek. Samotný nádor je zobrazený vo forme 2D a 3D modelov s farebným odlíšením anomálií. Súčasťou je aj modul na spracovanie chýb merania, pomocou ktorého je vypočítaná nová, spoľahlivejšia hodnota markera.

Kľúčové slová: mozog, nádor, detekcia, magnetická rezonancia, spracovanie chýb merania, marker

Abstract:

KVET, Michal: *Data transformation from the magnetic resonance imaging of the head for 3D visualisation* [Diploma thesis] – The University of Žilina. Faculty of management science and informatics; Department of Informatics. Tutor: prof. Ing. Karol Matiaško, PhD. – Qualification level: Engineer in study program Applied Informatics. Žilina: FRI ŽU in Žilina, 2012.

Research in a medical field presents new methods for diagnostics and treatment of the patient. Cancer belongs to the most serious diseases and can even cause death. The aim of the „Brain tumour detection“ project is to detect, locate anomalies inside the brain and to monitor the progress in time. 26 standard markers and unlimited amount of own markers defined in the application allow to detect wide range of tumours. Tumour is visualized in 2D, 3D models to show everything strange coloured. It also allows to process measurement error to get more reliable value of the marker.

Keywords: brain, tumour, detection, magnetic resonance imaging, error processing, marker

Prehlásenie

Prehlasujem, že som diplomovú prácu spracoval samostatne pod odborným vedením vedúceho záverečnej práce a že som uviedol všetky použité pramene a literatúru, z ktorých som čerpal.

V Prievidzi, dňa 25. apríla 2012

Podpis:

Pod'akovanie

Úprimne ďakujem vedúcemu diplomovej práce prof. Ing. Karolovi Matiaškovi, PhD. za jeho odborné vedenie, cenné pripomienky a čas, ktorý mi poskytol pri vypracovaní záverečnej práce.

Moje pod'akovanie patrí taktiež kolegovi Bc. Jánovi Mešinovi za tvorivé nápady a spoluprácu počas projektovej výučby.

Napokon chcem poďakovať rodičom a súrodencom za ich neustálu podporu počas celého štúdia.

Obsah

Úvod.....	8
1 Analýza	9
1.1 Medicínska analýza.....	10
1.1.1 Vnútroľbkové nádory	10
1.1.1.1 Rozdelenie tumorov	10
1.1.1.2 Diagnostika	10
1.1.1.3. Liečba a prognóza.....	11
1.1.2 Magnetická rezonancia (MRI)	12
1.1.2.1 Fyzikálny princíp MRI.....	12
1.1.2.2 Využitie MRI v klinickej praxi	14
1.1.3 Markery	15
1.2 Analýza pomocou programového vybavenia a používané technológie.....	16
1.2.1 Syngo fastView	16
1.2.2 Štandard DICOM.....	17
1.2.3 Existujúce štúdie venujúce sa biomedicínskym aplikáciám zameraným na spracovanie obrazu.....	19
1.2.3.1 Spracovanie statickej a dynamickej obrazovej informácie pre biomedicínske aplikácie	19
1.2.3.2 3D model hlavy.....	21
1.2.4 Systémy spracovania obrazu.....	22
1.2.5 Metódy na rozpoznávanie obrazu	22
1.2.5.1 Neurónová sieť.....	23
1.2.5.2 Algoritmus k-najbližších susedov.....	25
1.2.5.3 Adaboost (Adaptive boosting)	26
1.2.6 Modely farbenia	26
1.2.6.1 Modely farbenia závislé na zariadení	26
1.2.6.2 Modely farbenia nezávislé na zariadení.....	29
2 Analýza požiadaviek.....	31
3 Riešenie a implementácia	32

3.1 Moduly, class diagramy, popis tried.....	32
3.1.1 Trieda <i>Parametre</i> definuje aktuálne nastavenie aplikácie:	34
3.1.2 Trieda <i>Farby</i>	35
3.1.3 Trieda <i>Kocka</i>	36
3.2 Spracovanie údajov s vysokou hodnotou chyby	43
3.3 Nastavenie aplikácie	47
3.4 Nastavenie aplikácie v režime „Admin“	48
3.4.1 Šifrovací algoritmus MD5	48
3.4.1.1 Narodeninový útok	49
3.5 Zobrazovacie módy aplikácie	50
3.5.1 Grafy	52
3.5.2 2D modely.....	53
3.5.3 3D model.....	57
3.6. Porovnanie výsledkov z rôznych období	58
3.7 Use Case	59
4 Výsledky	62
5 Záver	64
Zoznam použitej literatúry a iných zdrojov	65
Knižné zdroje	65
Internetové zdroje	66
Zoznam použitých skratiek	68
Zoznam obrázkov	69
Zoznam rovníc	71
Zoznam príloh.....	Chyba! Záložka nie je definovaná.

Úvod

Výskum v ktorejkoľvek oblasti života je dôležitý, pretože vďaka nemu ľudská spoločnosť neustále napreduje. Špecifikom medicínskeho vývoja a výskumu je to, že sa svojím spôsobom dotýka každého z nás. Aplikáciou nových poznatkov či technických novínok do bežnej medicínskej praxe môžeme rozšíriť diagnostické alebo terapeutické možnosti, čím sa dá ovplyvniť osud konkrétneho pacienta.

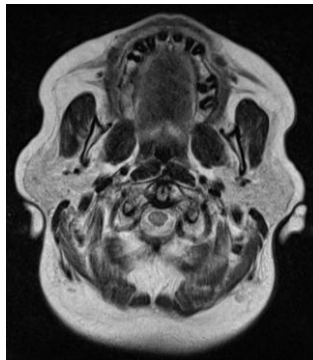
Onkologické ochorenia sú jedným z najväznejších problémov, pred ktorým stojí súčasná medicína. Svojou závažnosťou ovplyvňujú predovšetkým život pacienta, ale aj jeho okolia a celej spoločnosti. V centre záujmu mnohých odborníkov – od lekárov cez molekulárnych biológov až po IT špecialistov zameraných na problematiku medicínskeho vývoja stojí práve problematika včasnej diagnostiky, správnej liečby a vyhľadávanie rizikovej populácie. Adekvátne liečba včas diagnostikovaného ochorenia dáva pacientovi možnosť uzdravenia sa, zlepšenia kvality života, návrat do bežného života a samozrejme aj zníženie nákladov na liečbu.

Medicínsky orientovaný je aj inžiniersky projekt Nemocničný informačný systém na Fakulte riadenia a informatiky Žilinskej univerzity pod vedením prof. Ing. Karola Matiaška, PhD. Základom projektu je spolupráca s Fakultnou nemocnicou v Martine, pre ktorú je informačný systém vytváraný a praktické požiadavky používateľov informačného systému sú transformované do softvéru.

Cieľom našej časti projektu a teda aj tejto diplomovej práce, je detegovať, lokalizovať a zobraziť nádor mozgu, jeho polohu a vlastnosti. Lekár, pre ktorého je tento projekt určený, tak môže jednoduchšie interpretovať výsledky magnetickej rezonancie (MRI). Všetky anomálie sú na základe meraní farebne zobrazené a modelované. Naša aplikácia ponúka rôzne druhy zobrazenia - 3D model hlavy, 2D model rezov a grafy. Prináša aj možnosť výpočtu nedefinovaných, resp. nesprávne meraných miest v mozgovej časti hlavy. Metodika by mohla byť perspektívne využitá aj v diagnostike iných ochorení mozgu.

1 Analýza

Inžiniersky projekt Nemocničný informačný systém vznikol v roku 2009 ako začiatok dlhodobej spolupráce Fakulty riadenia a informatiky Žilinskej univerzity a Univerzitnej nemocnice v Martine. Cieľom tohto projektu je vyvíjať a vylepšovať programové vybavenie pre potreby pracovníkov nemocnice na základe ich požiadaviek a pripomienok. Tento projekt je teda priamo prepojený s praxou. V prvotnej fáze bola vytvorená databázová a modulárna štruktúra pre ambulancie za účelom vytvorenia jednotnej komunikácie, spracovania a ukladania údajov. V septembri 2010 sa v rámci projektu vytvorila nová programátorská skupina venujúca sa detekcii nádoru mozgu, výsledkom ktorej je aj táto diplomová práca. Pôvodnou úlohou malo byť rozpoznávanie a vizualizácia anomálií mozgu na základe DICOM súborov (štandard popísaný nižšie), resp. rezov mozgu v rôznych rovinách. Keďže výstupom vyšetrenia magnetickou rezonanciou sú aj hodnoty látok svedčiace o prítomnosti látok - markerov (metabolitov) v jednotlivých miestach, úlohou bolo tieto hodnoty spracovať, príp. odstrániť, resp. nahradiť nesprávne (chybné) hodnoty meraní a tieto výsledky poskytnúť medicínskym pracovníkom v zrozumiteľnej forme vo forme grafov, 2D a 3D modelov. Rezy mozgu vo forme obrázkov totiž neobsahujú až toľko informácií, zobrazujú len vrstvy v mozgu, bez zobrazenia hodnôt markerov z meraní. Kritickým faktorom je kvalita samotného obrazu, ktorá ovplyvňuje jeho informačnú hodnotu. Aj nepatrný pohyb pacienta počas vyšetrenia môže spôsobiť, že zobrazenie bude nekvalitné a výsledky môžu byť skreslené, príp. vôbec nepoužiteľné. Existujú metodiky, ako tieto artefakty odstrániť, ide však o zložitý proces úpravy zobrazeného nálezu s často neistým výsledkom. To bol aj dôvod spracovania jednotlivých metabolitov a markerov. Hlavným dôvodom pre vývoj bol fakt, že hodnoty boli doteraz spracovávané len „ručne“ bez možnosti tvorby nových markerov, vizualizácie a hodnotenia.



Obrázok 1 – Ukážka rezu hlavy

1.1 Medicínska analýza

Orientácia inžinierskeho projektu Nemocničný informačný systém na prax vyžaduje aj určitý náhľad na medicínsku problematiku, s ktorou projekt priamo súvisí. Vývoj predkladanej aplikácie konkrétne vyžadoval základný prehľad o problematike nádorov mozgu a zobrazovacích metód využívaných v ich diagnostike.

1.1.1 Vnútrolebkové nádory

Vnútrolebkové nádory sú rôznorodou skupinou nádorov, ktoré sa líšia svojou lokalizáciou, príznakmi, histologickým zložením a výskyt jednotlivých druhov závisí predovšetkým od veku. Vo všeobecnosti možno povedať, že nádory mozgu spôsobujú príznaky, ktoré sú vyvolané poškodením funkcie mozgu a mozgových nervov a za určitých okolností aj zvýšením vnútrolebkového tlaku. Najčastejšími príznakmi sú tzv. zánikové príznaky, ktorými sú poruchy hybnosti končatín, poruchy citlivosti, zraku, reči alebo psychické zmeny. Ďalšou skupinou príznakov sú príznaky vyplývajúce z lokálneho dráždenia mozgového tkaniva prejavujúce sa ako rôzne druhy epileptických záchvatov. Ako tzv. syndróm zvýšeného vnútrolebkového tlaku sa označuje súbor príznakov, ku ktorým patrí predovšetkým bolesť hlavy, zvracanie a poruchy zraku. [26] [17]

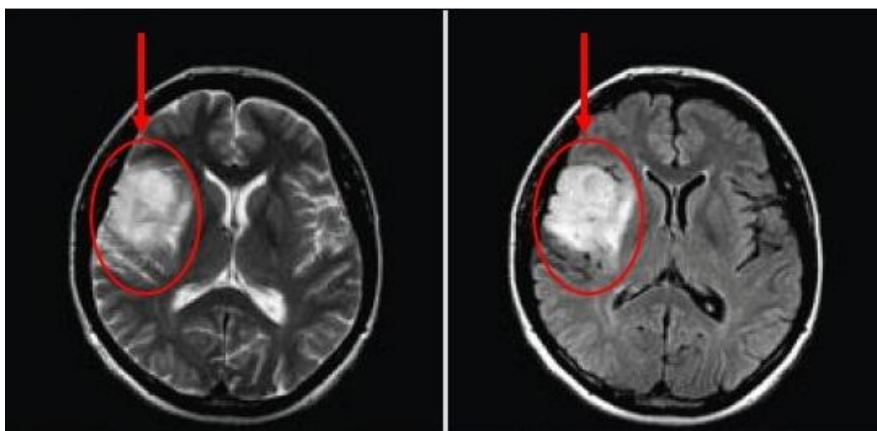
1.1.1.1 Rozdelenie tumorov

Na základe histologickej skladby rozdeľujeme tumory do niekoľkých skupín. Zjednodušene možno vnútrolebkové nádory rozdeliť na nádory vychádzajúce z mozgových buniek, z mozgových nervov, z mozgových obalov, nádory vychádzajúce z krvných buniek, nádory podmozgovej žľazy, nádory prerastajúce z okolia a nádory, ktoré sa do mozgu dostali metastatickým procesom, teda šírením nádorových buniek krvným riečiskom z miesta primárneho nádoru (najčastejšie nádor hrubého čreva, pľúc, prsníka). Rozdelenie tumorov na zhubné a nezhubné má relatívne menší význam ako pri nádoroch iných orgánov, nakoľko v mozgu pevne uzavretom v lebke môže aj pomaly rastúci nezhubný nádor spôsobiť závažné ohrozenie zdravia a života pacienta poškodením mozgu tlakom rastúceho nádoru. [26] [17] [36]

1.1.1.2 Diagnostika

Nádory mozgu sa spravidla diagnostikujú až potom, ako sa objavia príznaky poškodenia mozgových funkcií. Základnými diagnostickými metódami sú zobrazovanie

metódy: predovšetkým počítačová tomografia (CT) založená na princípe röntgenových lúčov, magnetická rezonancia (viď kapitola 1.1.2) a pozitronová emisná tomografia (PET) založená na princípe sledovania zvýšeného metabolizmu cukru nádorovými bunkami. Pomocnými vyšetreniami sú neurologické, ušné, očné vyšetrenie, krvné testy, vyšetrenie mozgovomiechového moku a ďalšie špecializované vyšetrenia. Definitívnu diagnózu možno stanoviť na základe stereotaktických neurochirurgických metód, ktoré umožňujú odobratie vzorky z tumoru tenkou ihlou navádzanou pod CT za účelom histologického vyšetrenia. [26] [17] [36]



Obrázok 2 - Nádor mozgu - anaplastický astrocytóm [44]

1.1.1.3. Liečba a prognóza

Najbežnejším spôsobom liečby mozgových nádorov je chirurgický výkon. Pri nezhubných nádoroch slúži ako jediný terapeutický postup. Aj keď niekedy nádor nemožno vyliečiť, aj jeho čiastočné odstránenie môže zmierniť prejavy poškodenia mozgu. Okrem úplného alebo čiastočného odstránenia nádoru poskytuje chirurgický zákrok aj vzorky na histologické vyšetrenie. Na základe histologického nálezu sa stanoví definitívna diagnóza a určí sa vhodný typ ďalšej liečby. Chirurgické výkony môžu byť limitované lokalizáciou nádoru v blízkosti dôležitých oblastí mozgu, ciev alebo v ťažko prístupných miestach.

Ďalšou liečebnou metódou je rádioterapia, teda ožarovanie tumoru röntgenovými lúčmi alebo prúdom nabitých častíc. Pri vonkajšom ožarovaní sa lúče z ožarovacieho prístroja mimo tela pacienta nasmerujú do miesta tumoru. Pri stereotaktickej rádiouchirurgii sa dávka žiarenia zamieri podľa CT alebo MRI nálezu priamo do lôžka nádoru, čím sa čiastočne chráni okolité tkanivo. Používajú sa pritom rôzne technické zariadenia typu gama-noža. Metódou vnútornej rádioterapie je brachyterapia, pri ktorej sa rádioaktívna látka vo forme zŕn implantuje priamo do miesta tumoru.

Tretou možnosťou liečby je chemoterapia, ktorá používa na zničenie nádorových buniek lieky - cytostatiká. Cieľom chemoterapie môže byť zničenie pozostatkov tumoru po chirurgickom výkone, spomalenie rastu neoperovateľného tumoru alebo zlepšenie stavu pacienta s ústupom jeho ťažkostí a predĺžením života.

Prognóza pacienta s vnútrolebkovým nádorom závisí od mnohých faktorov – od typu, lokalizácie a histologického zloženia nádoru, od miery jeho operovateľnosti, veku pacienta, jeho celkovej kondície a pridružených ochorení, od možností liečby a reakcie tumoru na podávanú liečbu. [26] [17] [36]

1.1.2 Magnetická rezonancia (MRI)

Zobrazovanie magnetickou rezonanciou je diagnostická metóda používaná na vytváranie dvojrozmerných a trojrozmerných obrazov orgánov ľudského tela. Metóda, ktorá sa v zdravotníctve používa približne od roku 1980, je jedinečná a čoraz častejšie využívaná predovšetkým pre jej schopnosť vytvoriť kvalitný obraz bez použitia radiačného žiarenia resp. zobrazit' aj štruktúry ľudského tela, ktoré iné metódy nedokázali dostatočne zobrazit'. Technologický pokrok prináša do kvality zobrazovania neustále inovácie, čím sa diagnostika ochorení zrýchľuje a presňuje.

1.1.2.1 Fyzikálny princíp MRI

Zobrazovanie magnetickou rezonanciou je založené na princípe zisťovania zmien magnetických momentov jadier prvkov s nepárnym protónovým číslom umiestnených v silnom statickom magnetickom poli po aplikácii rádiových pulzov. V dôsledku rotácie jadier atómov okolo svojej osi (spin) vzniká okolo týchto jadier magnetické pole (magnetický moment). Atóm vodíka obsahuje v jadre jediný protón a keďže sa v tkanivách vyskytuje v hojnom množstve, využíva sa v MRI diagnostike. Ak vložíme skúmané tkanivo do silného magnetického poľa, dôjde k usporiadaniu spinov protónov do jedného smeru. V skutočnosti ide o dva opačné smery, z ktorých jeden prevažuje a výsledný magnetický moment je teda v jednom smere. V tomto stave koná magnetický moment protónov dva druhy pohybov – jednak rotuje okolo svojej osi (spin) a zároveň sa pohybuje po plášti pomyselného kužeľa, čo sa označuje ako precesia. Ak je na tkanivo aplikovaný rádiový pulz (teda elektromagnetické vlnenie v pásme rozhlasových krátkych vĺn) s frekvenciou zhodnou s frekvenciou precesie protónov, dôjde podľa princípu rezonancie k vychýleniu magnetických momentov z pôvodného smeru o určitý uhol a zároveň aj k synchronizácii precesie všetkých

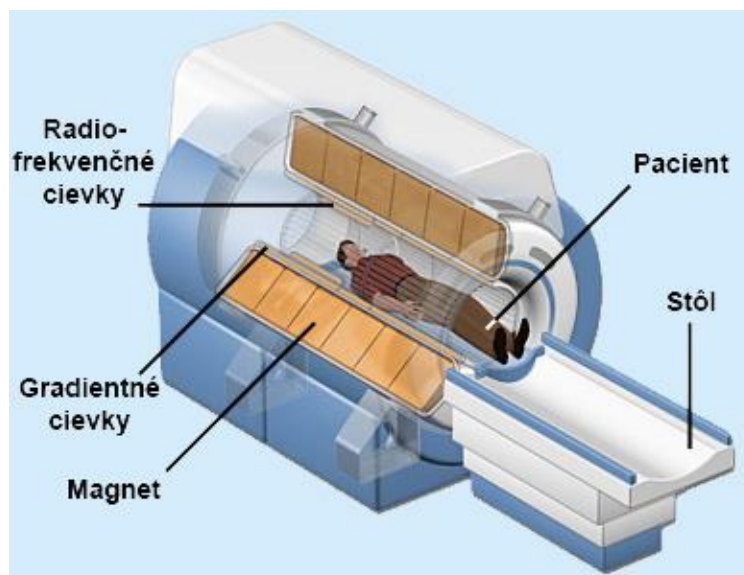
protónov. Po skončení pulzu dochádza postupne k návratu do pôvodného stavu. Čas, za ktorý k tom dôjde, označujeme ako relaxačný čas. Čas nutný k návratu vychýleného magnetického momentu označujeme ako relaxačný čas T1, „rozsynchronizovanie“ precesie označujeme ako relaxačný čas T2. Oba závisia predovšetkým od zloženia hmoty v okolí skúmaných protónov. Signál, ktorý získame po sérii rôznych rádiových pulzov, má rovnaký charakter - ide o elektromagnetické vlnenie, ktoré môžeme registrovať pomocou prijímacích cievok (v podstate antén) a merať jeho veľkosť. Séria pulzov nutná k získaniu merateľného signálu je označovaná ako sekvencia. Rezonančná frekvencia protónov je závislá od intenzity vonkajšieho magnetického poľa. Ak toto pole pomocou gradientových cievok vhodne upravíme, môžeme získať informácie o mieste, z ktorého prichádza signál (protóny na iných miestach s touto frekvenciou nerezonovali a teda nemôžu vydávať žiaden signál).

Princíp MRI vychádza z merania magnetických momentov jadier atómov vodíka vo vode a niektorých tukoch. Informácie o ostatných tkanivách získame nepriamo – inak sa správajú protóny vody v susedstve veľkých molekúl bielkovín a inak, ak sú obklopené iba molekulami vody.

Pri vyšetrení zhotovujeme vrstvom obrazy pomocou rôznych typov sekvencií, ktoré nás informujú o rozdieloch relaxačných časov T1 a T2 (teda T1 alebo T2 vážená sekvencia) alebo o množstve protónov (sekvencia vážená podľa protónovej denzity). Pre získanie úplnej informácie je treba porovnať intenzitu signálu rovnakého miesta pri rôznych typoch sekvencií.

V niektorých prípadoch sa využíva aj podanie kontrastnej látky, ktoré zabezpečí zobrazenie zmien, ktoré svojím magnetickým správaním vyvolá v okolitých molekulách vody. Najčastejšie sa používajú cheláty obsahujúce gadolínium, teda prvok zo skupiny lantanoidov, ktorý mení magnetické pomery vo svojom okolí.

Dĺžka vyšetrenia sa pohybuje rádovo v desiatkach minút (20-60 minút) v závislosti od rýchlosti prístroja a počtu zobrazených sekvencií. [26] [17] [5]



Obrázok 3 - Schéma MRI [45]

Pri popise vyšetrenia využívame termíny vyjadrujúce intenzitu signálu - hyperintenzívny (hypersignálny, na obrazoch svetlý), izointenzívny (izosignálny, s rovnakou intenzitou signálu), hypointenzívny (hyposignálny, na obrazoch tmavý) a asignálny (bez signálu, na obrazoch čierny). Rovnaké tkanivá majú v rôznych sekvenciách odlišnú intenzitu. Napr. tekutina je v T2 váženej sekvencii hypersignálna, v T1 hyposignálna. Vo všeobecnosti má tuk vysokú intenzitu signálu, naopak, kosť je vždy asignálna, lebo obsahuje len minimálne množstvo vody. [26] [17] [5]

1.1.2.2 Využitie MRI v klinickej praxi

MRI sa najčastejšie používa v neurorádiológii, teda pri zobrazovaní ochorení mozgu, miechy a chrbtice. Okrem toho sa využíva na diagnostiku porúch pohybového aparátu – dajú sa zobraziť všetky jeho súčasti od kostí a kostnej drene cez väzy, šľachy, svaly a chrupavky až po tekutinu v kĺboch. Stále častejšie sa MRI využíva aj na zobrazenie orgánov v oblasti brucha, hrudníka či krku. Najčastejšie sa využíva na dôkaz alebo vylúčenie ložiskových zmien alebo tumorov. Výhodou MRI oproti CT je podrobnejšie zobrazenie mäkkých štruktúr, možnosť zobrazenia v ľubovoľnej rovine a absencia ionizujúceho žiarenia. MRI nemožno použiť u pacientov s kardiostimulátorom, zvýšiť opatrnosť treba pri pacientoch s implantovanými kovovými materiálmi v tele, nakoľko hrozí ich zahriatie, resp. posun. [26] [17] [5]



Obrázok 4 - MRI prístroj (Siemens Avanto) [45]

1.1.3 Markery

Nádorový marker je látka, najčastejšie bielkovinovej povahy, ktorej výskyt poukazuje na prítomnosť nádoru v ľudskom organizme. Táto látka môže byť takmer výlučne produktom nádorových buniek, teda v zdravých bunkách sa takmer nevyskytuje, ale markerom môže byť aj normálna súčasť bunky, ktorej prítomnosť v abnormálnom množstve alebo abnormálnom čase (napríklad látka prítomná normálne počas embryonálneho vývoja znovu sa objavujúca v tkanive dospelého človeka) rovnako svedčí pre možný zhubný proces. Vo všeobecnosti môžeme sledovať prítomnosť týchto markerov buď priamo v tkanive, čím môžeme odlíšiť normálne tkanivo od chorobne zmeneného, alebo v telových tekutinách (napr. sérum, moč, mozgovomiechový mok), do ktorých sa tieto látky uvoľňujú z tkanív. Pre potreby tejto diplomovej práce používame termín marker pre látku, ktorá sa vyskytuje priamo v nádorovom tkanive. Markerom v tomto slova zmysle môžu byť buď bielkoviny na povrchu bunky alebo bielkoviny, ktoré sú súčasťou metabolizmu buniek (napr. enzýmy, hormóny). Osobitnou skupinou sú markery, ktoré vznikajú pri odumieraní a rozpade nádorových buniek (napr. fragmenty cytoplazmových štruktúr). Väčšina nádorov je charakteristických prítomnosťou jedného alebo niekoľkých viac či menej špecifických markerov. Sledovaním prítomnosti viacerých markerov – aj keď s nižšou senzitivitou a špecificitou – získavame celkový obraz o produkcii týchto látok konkrétnym nádorom.

Pri vyhľadávaní nádorových ochorení v populácii sa sledovanie prítomnosti markerov v krvi takmer nepoužíva, nakoľko tieto vyšetrenia majú pomerne vysokú senzitivitu, ale nižšiu

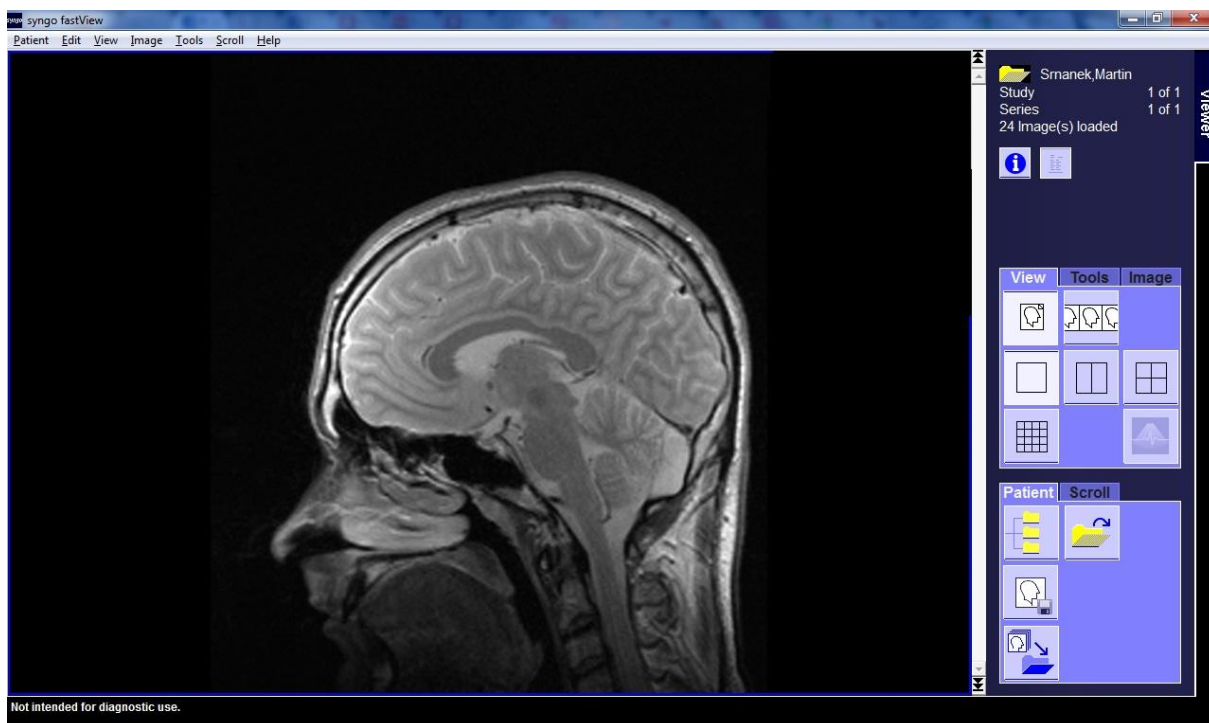
špecifickosť. Význam sledovania prítomnosti nádorových markerov je naopak zásadný u pacientov, u ktorých je potvrdená prítomnosť zhubného nádoru. V čase stanovenia diagnózy má význam poznať vstupné hodnoty markerov tak v krvi ako aj v nádorovo zmenenom tkanive a to jednak pre zaradenie pacienta do liečebného protokolu a aj pre určenie predpokladanej prognózy. Získa sa tak akýsi biochemický a imunologický „profil“ nádoru pred začatím liečby. Prítomnosť niektorých markerov v krvi alebo tkanive svedčí pre agresívny poddruh nádoru s horšou prognózou, prítomnosť iných markerov môže byť naopak dobrým prognostickým znamením. Informácie o význame jednotlivých markerov pre konkrétne typy nádorov boli získané štatistickým sledovaním veľkých súborov pacientov. Sledovanie prítomnosti týchto markerov má význam aj pri hodnotení efektu liečby. Pokles markerov svedčí pre dobrý efekt liečby. Ak sa hodnoty nemenia alebo stúpajú, svedčí to o rezistencii ochorenia na konkrétnu liečbu. Sledovanie prítomnosti markerov sa využíva aj u pacientov po ukončení onkologickej liečby. Ak hodnoty začnú stúpať, znamená to, že počet nádorových buniek v organizme opäť rastie, čo svedčí buď pre znovuoobjavenie sa nádoru alebo o rozšírenie nádoru vo forme metastáz a to aj v prípade, že pacient ešte nemá žiadne nové ťažkosti. Na základe toho sa môže pristúpiť k liečbe čo najskôr.

1.2 Analýza pomocou programového vybavenia a používané technológie

V súčasnosti používané aplikácie sa zameriavajú hlavne na zobrazovanie výsledkov magnetickej rezonancie vo forme obrazu. Ich hlavnou nevýhodou je fakt, že ide len o prehliadače a teda nedávajú priamo lekárovi žiadnu informáciu o existencii a polohe anomálie. Typickým príkladom je aj Syngo fastView.

1.2.1 Syngo fastView

Syngo fastView je aplikácia vyvíjaná spoločnosťou Siemens od roku 2004 na prehliadanie výsledkov magnetickej rezonancie (MRI). Jedná sa o prehliadač DICOM súborov. Jednotlivé rezy dokáže spracovávať synchrónne a vytvára tak dojem videa.



Obrázok 5 - Syngo fastView

1.2.2 Štandard DICOM

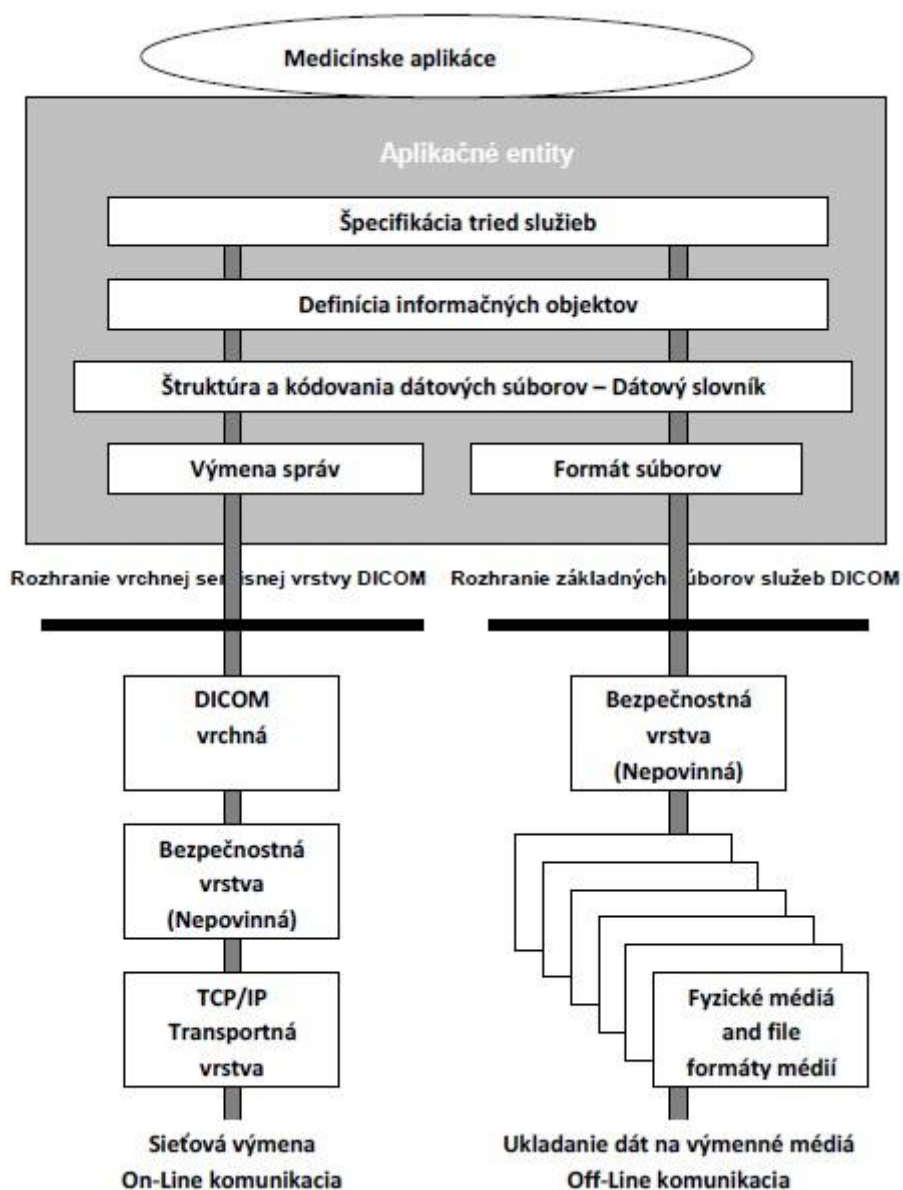
DICOM (Digital Imaging and Communications in Medicine) je štandard pre vytváranie, spracovanie, ukladanie a tlač medicínskych informácií. Je navrhnutý pre výmenu informácií, snímok a dokumentov medzi aplikáciami, ako aj prístrojmi používanými v oblasti medicíny. Súčasná štruktúra bola štandardizovaná v roku 1993.

V súčasnosti je štandard určený pre digitálne spracovanie a prenos obrazu v medicíne a bol definovaný spoločnosťami ACR (American College of Radiology) a NEMA (National Electrical Manufacturers Association).

Základné požiadavky na štandard DICOM:

- Podpora sieťovej komunikácie pomocou protokolu TCP/IP.
- Definovanie štruktúry dát pre posielanie iných aplikáciám – štruktúra DICOM sa skladá z viacerých častí, čo umožňuje jej ďalší vývoj pridaním nových funkcií. Návrh používa smernicu ISO definujúcu štruktúru dokumentov - sada protokolov pre vyhlásenie zhody zariadení zo štandardom.
- Určovanie sémantiky služieb, príkazov a dát, formáty adresárov a súborov.

V súčasnosti je tento štandard stále vyvíjaný a vylepšovaný. Zastrešuje ho Komisia DICOM na základe informácií od používateľov údajov definovaných štandardom DICOM. Nasledujúci obrázok zobrazuje základný komunikačný model definovaný štandardom DICOM.[37] [38] [39]



Obrázok 6 - Základný komunikačný model DICOM [39]

1.2.3 Existujúce štúdie venujúce sa biomedicínskym aplikáciám zameraným na spracovanie obrazu

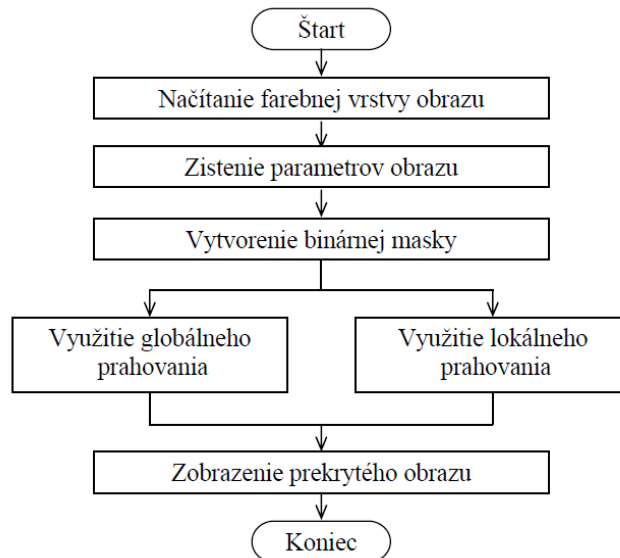
Samotný obraz je možné zobrazit' alebo ďalej spracovať s cieľom úpravy kvality obrazu, získania ďalších informácií, napr. detekcia objektov záujmu. Využíva sa široké spektrum analytických metód – lokálne a globálne prahovanie, entropia a pod.

1.2.3.1 Spracovanie statickej a dynamickej obrazovej informácie pre biomedicínske aplikácie

Habilitačná práca doc. Ing. Libora Hargaša, PhD. z Elektrotechnickej fakulty Žilinskej univerzity [4] sa venuje princípom spracovania obrazu pre potreby medicíny, konkrétne zobrazovaniu prekrvenia kože, anomáliám, výpočtom a zobrazením plochy, ktorá nie je dostatočne prekrvená s cieľom poukázať na patologické zmeny. Výsledky spracovania sa používajú pri vyšetrení srdca a prietoku cievami.

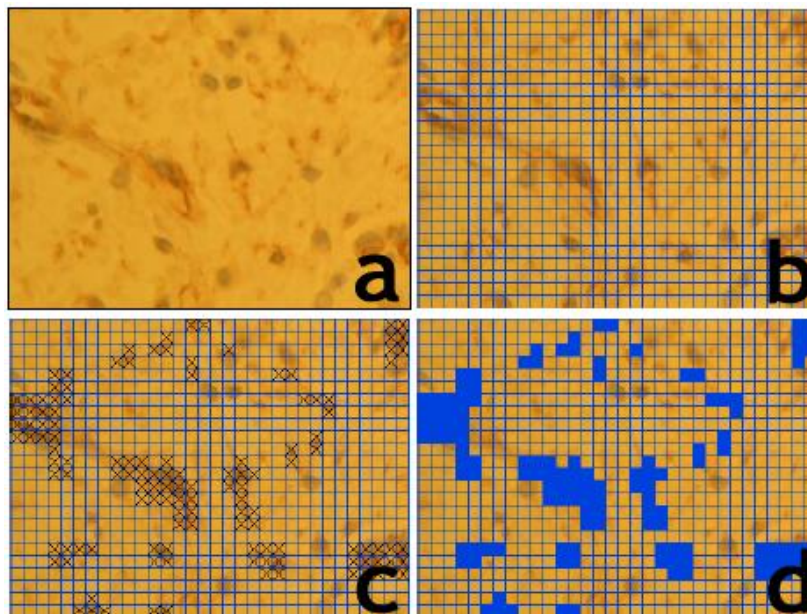
Medzi základné prístupy spracovania a analýzy obrazu patrí globálne prahovanie, ktoré sa využíva hlavne pri snímkach, ktoré sú vytvorené pri rovnakých svetelných podmienkach. Medzi základné techniky globálneho prahovania patrí clustering (rozdelenie obrazu do viacerých kategórií na základe histogramu), entropia (detekcia veľmi malých objektov využívaná hlavne pri detekcii chýb), rozdiel medzi triedami (interclass variance). Táto metóda sa snaží zaradiť obraz do tried tak, aby rozdiely medzi jednotlivými triedami boli čo najväčšie. Globálne prahovanie vytvára šedotónový obraz, resp. binárny, oddeľujúci objekty záujmu a pozadie (background).

Lokálne prahovanie nepoužíva jednotný prah pre celý obraz, ale zaraďuje pixely do kategórií podľa hodnôt okolitých pixelov (okolie). Využíva sa v prípadoch, keď obraz nie je rovnomerne osvetlený. Algoritmus porovnáva hodnotu lokálneho kritéria (vypočítanú z okolia) a pôvodnú hodnotu pixelu. Tým oddeľuje pozadie od objektu záujmu. Veľkosť okolia môže byť definovaná automaticky alebo používateľom. Tento parameter však výrazne ovplyvňuje dobu spracovania - čím je okolie väčšie, tým je čas spracovania väčší. Nasledujúci diagram zobrazuje algoritmus nastavenia prahu.



Obrázok 7 - Diagram algoritmu pre nastavenie prahu [4]

Nasledujúca séria obrázkov zobrazuje princíp spracovania a lokalizácie ciev. Na nasnímanom obraze (obr. 8a) sa vytvorí raster (obr. 8b), ktorý je spracovaný „krížikovou metódou“ (obr. 8c). Výsledkom je poloha a plocha cievneho pokrytia (obr. 8d). Vypočítaná plocha je neskôr štatisticky vyhodnotená s cieľom určenia a lokalizácie patologických zmien.



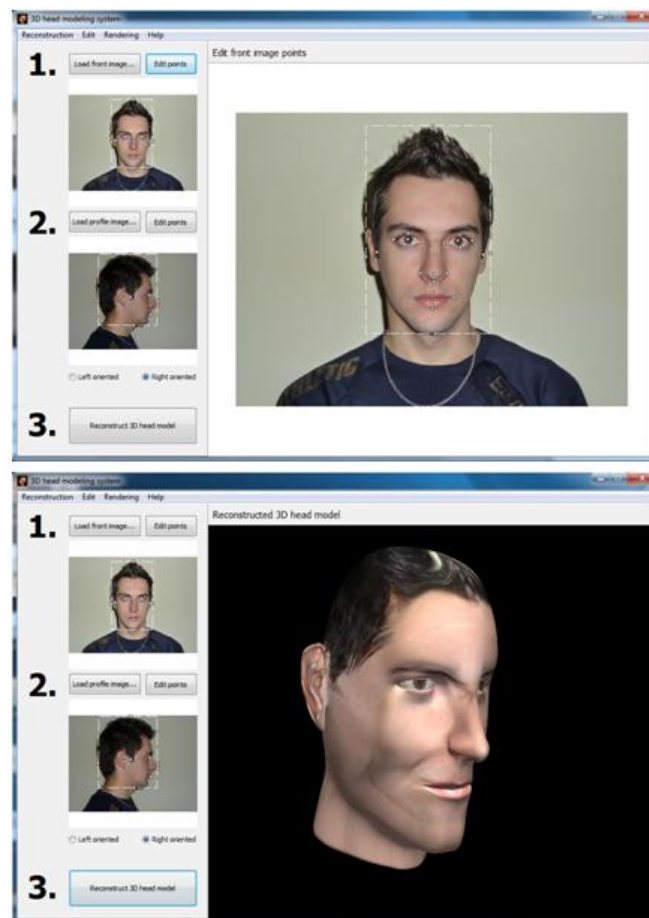
Obrázok 8 - Lokalizácia cievneho pokrytia [4]

1.2.3.2 3D model hlavy

Cieľom práce „Automatic image-based 3D head modeling“ [8] je navrhnúť transformáciu 2D fotografií (hlava človeka spredu a z boku) na 3D model. Využíva sa parametrizovaný model podľa detegovaných parametrov z fotografií. Tento program poskytuje možnosť vytvorenia trojdimenzionálneho modelu hlavy s rýchlym a kvalitným výsledkom. Využíva sa spracovanie obrazu a techniky počítačového videnia.

Táto práca dala základ pre vytvorenie 3D modelu hlavy.

Nasledujúce obrázky zobrazujú výsledky spracovania a 3D model vytvorený aplikáciou, ktorá je voľne dostupná na webovej adrese <http://www.peterkan.com/diplomathesis.php> [27].



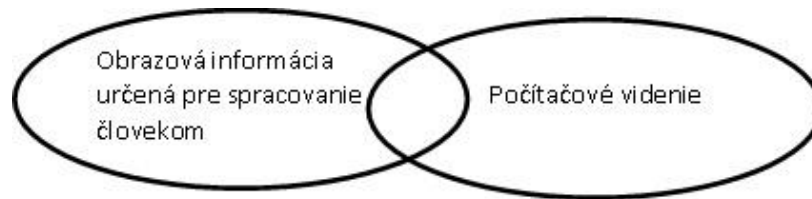
Obrázok 9 - 3D model hlavy [27]

1.2.4 Systémy spracovania obrazu

Význam spracovania digitálneho obrazu môžeme v zásade rozdeliť na dve oblasti:

- spracovanie obrazu do tvaru určeného pre vizuálne vnímanie človekom,
- oblasť počítačového videnia.

Na prvý pohľad by sa mohlo zdať, že tieto oblasti sú oddelené a medzi sebou nesúvisia. Opak je však pravdou, tieto časti sa dokonca prelínajú. Existujú oblasti a aplikácie, kde potrebujeme obidva spôsoby spracovania. V tom prípade však treba brať ohľad na obmedzenia a rozdielnosti jednotlivých typov. Úlohou oboch typov je spracovať a upraviť obraz tak, aby jeho informačná (interpretovaná) hodnota bola čo najvyššia.



Obrázok 10 - Typy spracovania obrazovej informácie

Typickými príkladmi úpravy pre vizuálne spracovanie človekom sú satelitné a letecké snímky, využitie nájdu aj v oblasti architektúry, reštaurátorstve, astronómii, kartografii či robotike. V oblasti medicíny je to spracovanie výsledkov zobrazovacích vyšetrení – röntgen, CT a MRI.

Počítačové videnie je preferované v prípadoch, kedy by spracovanie človekom bolo náročné a dlhotrvajúce. Medzi aplikácie počítačového videnia patrí kontrola kvality produktov, rozpoznávanie znakov, v oblasti medicíny je to spracovanie a porovnávanie DNA. [28] [29]

1.2.5 Metódy na rozpoznávanie obrazu

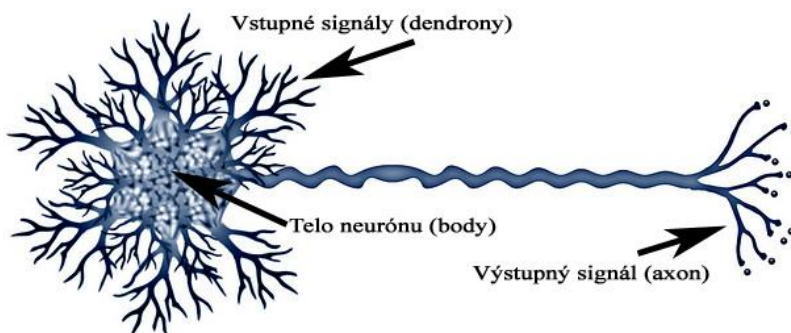
Úspešná detekcia objektov lokalizovaných v obraze nie je jednoduchý problém. Existuje veľké množstvo algoritmov a techník, ktoré majú špecifické vlastnosti ovplyvňujúce ich použitie.

1.2.5.1 Neurónová sieť

Neurónová sieť je charakterizovaná ako masívne paralelný procesor. Bola vytvorená na základe sledovania činnosti ľudského mozgu. Napodobňuje ho v dvoch hľadiskách [30]:

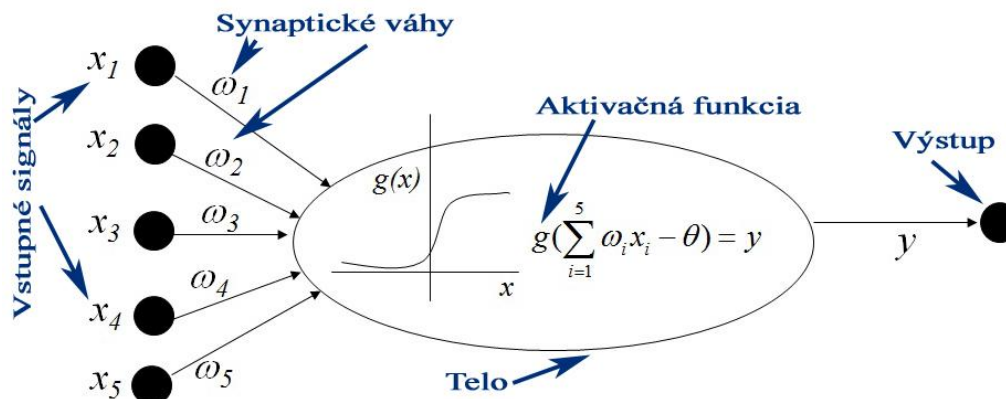
- poznatky sú zbierané v sieti počas učenia,
- medzineurónové spojenia (synaptické váhy) sú využívané na ukladanie znalostí.

Prvá neurónová sieť bola vytvorená v roku 1943 Warrenom McCullochom a Walterom Pitsom ešte pred masívnym začatím používania výpočtovej techniky. Je inšpirovaná biologickou neurónovou sieťou, napodobňuje niektoré procesy v mozgu. S príchodom počítačov sa vývoj zrýchlil, v roku 1952 bola vydaná kniha Design of a Brain: The Origin of Adaptive Behavior (dr.Ashby), ktorá mala výrazný vplyv na ďalší rozvoj neurónových sietí. [31] [9]



Obrázok 11 - Neurón

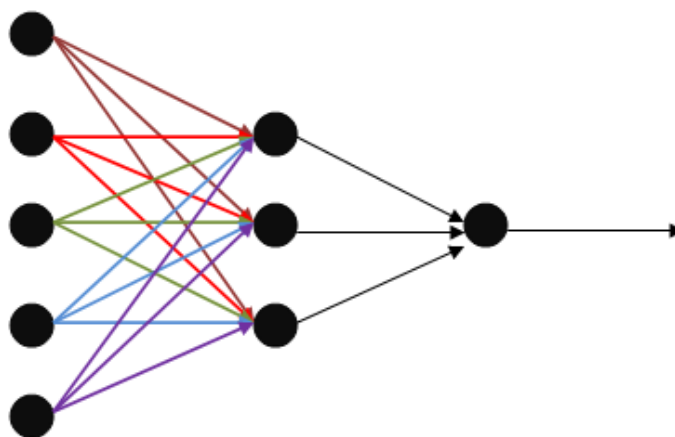
Základom neurónovej siete sú neuróny. Každý neurón má viac vstupov ($x_1, x_2, \dots, x_n$) a každý z nich je ohodnotený synaptickou váhou (dôležitosťou daného vstupu). Vstupy a váhy sú privádzané do aktivačnej funkcie, ktorej výstup je odovzdaný v axóne (výstup neurónu). Nasledujúci obrázok zobrazuje štruktúru neurónu použitého v neurónovej sieti.



Obrázok 12 - Matematická formulácia neurónu

Na vstup neurónu je privedený signál (informácia). Jeho hodnoty sú vynásobené synaptickými váhami (ω_i). Súčet násobkov synaptických váh a vstupných signálov je po odpočítaní prahu (threshold - θ) privedený ako vstup do aktivačnej funkcie. Prahová hodnota definuje minimálnu hodnotu aktivácie neurónu.

Obr. 13 zobrazuje neurónovú sieť – viacvrstvový perceptrón. Skladá sa zo vstupnej vrstvy (naľavo) – input layer, nasledujú skryté vrstvy (hidden layer) a výstupná vrstva (output layer) (napravo). Model zobrazuje doprednú sieť, teda výstupy vrstvy sú privedené na vstupy nasledujúcej vrstvy. Každý výstup neurónu jednej vrstvy je privedený na vstup každého neurónu nasledujúcej.[23] [24]



Obrázok 13 - Neurónová sieť

Zjednodušene možno povedať, že cieľom neurónovej siete je simulovať činnosť mozgu. Využíva sa pri rozpoznávaní vo veľmi zložitom systéme, ktorého matematický popis je takmer nemožný, resp. príliš zložitý. Pokiaľ však máme dostatočne veľkú vzorku údajov,

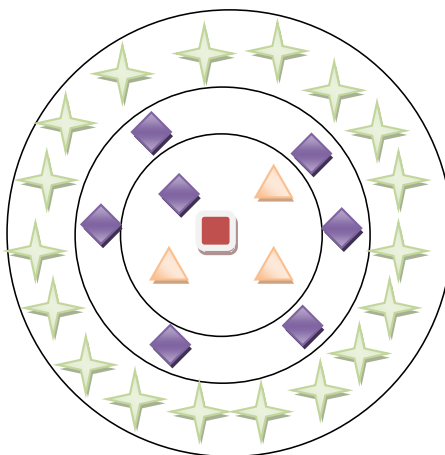
ktoré vstupujú do systému a k nim odpovedajúce požadované výstupy (odpovede), môžeme použiť neurónovú sieť. Vzorka údajov je použitá ako trénovacie údaje v procese učenia neurónovej siete. Proces učenia je zložitý proces, pri ktorom sa nastavujú synaptické váhy jednotlivých neurónov, jeho úlohou je zabezpečiť, aby rozdiely medzi definovanými výstupmi (predlohy) a výstupmi z neurónovej siete boli čo najmenšie.

V súčasnosti je možné vytvoriť omnoho rýchlejšiu neurónovú sieť, ako je biologická neurónová sieť. Podľa výskumov, ľudský mozog sa skladá z 10^3 až 10^4 neurónov. [30] Takýto rozsiahly systém však ešte nasimulovať nevieme. Môžeme teda tvrdiť, že hoci dokážeme vytvoriť rýchlejšiu neurónovú sieť, nedokážeme ňou riešiť tak zložité procesy ako ľudský mozog.

1.2.5.2 Algoritmus k-najbližších susedov

Algoritmus najbližších susedov (K-nearest neighbors) klasifikuje objekt na základe najväčšej zhody s trénovacími vzorkami. Patrí medzi najjednoduchšie algoritmy, klasifikuje vlastnosti objektu a zaraďuje ho do triedy na základe vlastností susedov. V prípade, že $k=1$, objekt je priradený do rovnakej triedy ako jeho najbližší sused.

Každý vstupný objekt je reprezentovaný bodom v N-rozmernom priestore. Ak chce klasifikátor určiť hodnotu atribútu v neznámej vzorke, hľadá v tomto priestore objekty, ktorých vzdialenosť (napr. Euklidovská, Manhattanovská) je čo najmenšia. Úlohou je správne zvoliť parameter k . Nasledujúci obrázok ukazuje princíp zaradenia do skupiny. V prípade, že zvolíme hodnotu $k=4$, neznámy objekt (červený štvorec) zaradíme do skupiny „trojuholník“. Ak hodnota parametra k bude 10, objekt zaradíme do kategórie „kosoštvorec“. V prípade, že $k=28$, objekt bude klasifikovaný ako „hviezdička“. [7] [32] [33]



Obrázok 14 - Algoritmus k-najbližších susedov

1.2.5.3 Adaboost (Adaptive boosting)

Algoritmus Adaboost patrí do kategórie algoritmov typu Boost - algoritmy, ktoré používajú viaceré klasifikátory pomocou výberu vzoriek z množiny tréovacích objektov. V súčasnosti je z tejto kategórie najpoužívanejší. Každý prvok dostane pridelenú váhu určujúcu pravdepodobnosť jeho výberu do tréovacej množiny. Ak algoritmus zaradí objekt do správnej triedy, jeho váha klesá. V opačnom prípade váha rastie. Algoritmus sa tak sústreďí hlavne na ťažko rozlíšiteľné vzorky.

Adaboost je zložený zo slabých klasifikátorov (weak classifier). Presnosť klasifikátora síce nie je vysoká, po spojení s viacerými však vytvorí silný klasifikátor (strong classifier) s výrazne lepšími vlastnosťami. Algoritmus Adaboost však trpí neprijemnou vlastnosťou – pretrénovanie. Vzniká zriedkavo pri veľkom počte požadovaných klasifikátorov. V tom prípade sa algoritmus príliš zameriava na jednotlivé črty, stráca tak vlastnosť zovšeobecňovania (generalizácie). [7] [1]

Medzi ďalšie algoritmy detekcie patria napríklad rozhodovacie stromy (decision trees) a podporné vektory (support vector machine).

1.2.6 Modely farbenia

Úspešnosť detekcie závisí nielen od architektúry rozpoznávania, ale aj od výberu dobrých príznakov, ktoré sú vstupmi (vektor príznakov) do rozpoznávacieho algoritmu.

Úlohou rozpoznávania je vyhľadať objekt záujmu, správne ho definovať a zaradiť. Dôležitou časťou je správne rozlíšiť hľadané objekty a „pozadie“. Model farbenia definuje základné farby a spôsob miešania farieb (aditívne, substraktívne). Typ modelu výrazne ovplyvňuje spôsob spracovania obrázka (niektoré algoritmy sú určené len pre niektoré farebné modely).

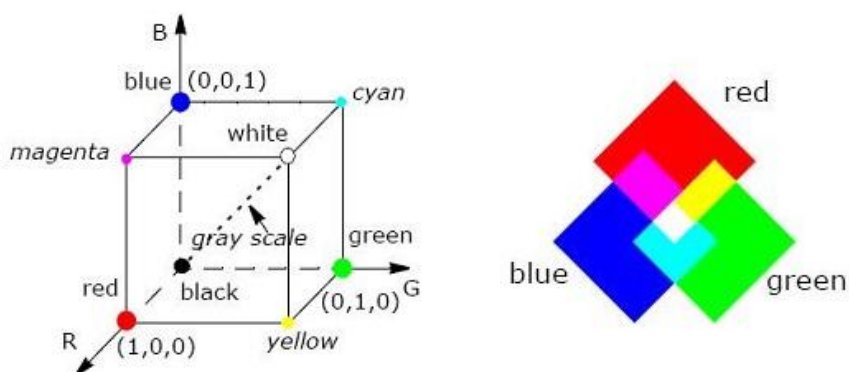
Rozlišujeme 2 základné skupiny modelov:

1.2.6.1 Modely farbenia závislé na zariadení

RGB & RGBA

Názov modelu vznikol z iniciálok troch základných farieb používaných v tomto modeli (R – red (červená), G – green (zelená), B – blue (modrá)). Každá farebná zložka môže mať ľubovoľnú intenzitu. Miešaním týchto základných farieb je možné vytvoriť akúkoľvek farbu spektra. Miešanie farieb je aditívne, teda miešaním farieb získavame svetlejšiu farbu. Celé spektrum je reprezentované 24 bitmi, každá farba odpovedá jednému bajtu (8 bitov) – existuje

teda 256 rôznych farebných variácií. Využíva sa hlavne v zobrazovacích zariadeniach (monitory, obrazovky televízorov,...). Tento model patrí do skupiny modelov závislých na zariadení - jednotlivé zariadenia môžu reprodukovat' farby rozdielne, intenzita farieb závisí od výrobcu a medzi jednotlivými zariadeniami sa líši.

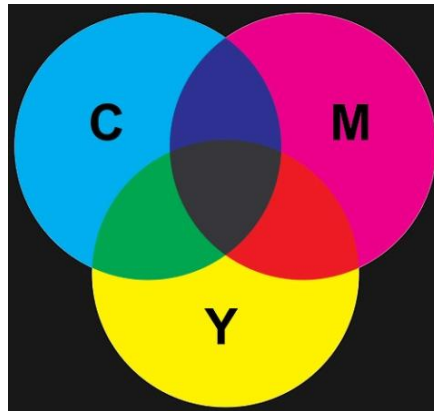


Obrázok 15 - RGB model [40]

Hlavnou nevýhodou spomínaného modelu je nemožnosť zobrazenia transparentnej farby. Tento nedostatok je vyriešený modelom RGBA, ktorého štruktúra je rovnaká ako pri modeli RGB, k štruktúre však pridáva ďalší bajt (alpha kanál) reprezentujúci priehľadnosť. [40] [3]

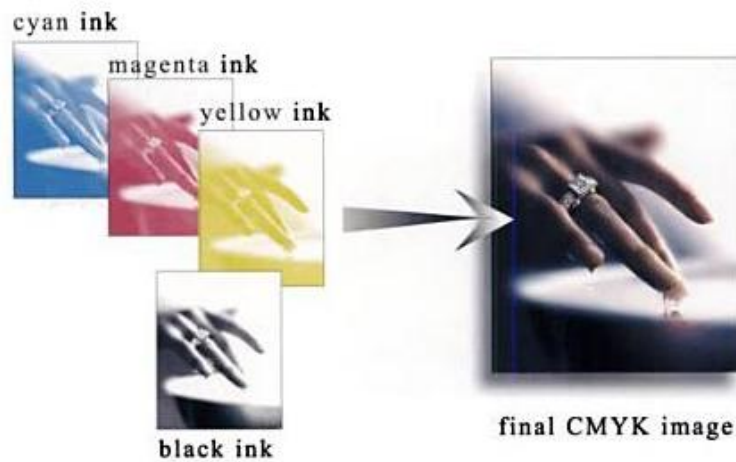
CMY & CMYk

Iným typom sú modely CMY a CMYk. Základom oboch sú tri farby – zelenomodrá (cyan), fialová (magenta) a žltá (yellow). Modely sú substraktívne, teda miešaním farieb dostávame tmavšiu farbu. Zmiešaním všetkých farieb pri reprezentácii modelom CMY však nezískavame čiernu farbu, ale tmavošedú (obr. 16). Pridaním samostatnej čiernej farby dostávame model CMYk (CMY + čierna farba (black)); keďže začiatkové písmeno čiernej farby (black) by mohlo reprezentovať taktiež modrú farbu, bolo zvolené posledné písmeno z názvu tejto farby – “k”).



Obrázok 16 - CMY model [6]

Model upravený o čiernu farbu je používaných v tlačovej technológii:

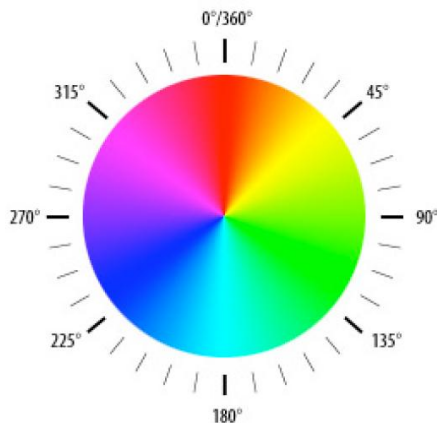


Obrázok 17 - Princíp tlače - CMYk model [6]

HSL & HSV

Modely HSL a HSV patria taktiež k základným modelom, reprezentácia je rovnaká ako vnímanie farieb ľudským okom. Oba sa skladajú z troch zložiek – odtieň, sýtosť a svetlosť (HSL), resp. jas pri modeli HSV. Vznikli z modelu RGB a sú ním podmienené:

Odtieň (hue) – popisuje farbu v rozsahu 0 – 360°. 0° reprezentuje červenú farbu, postupne prechádza cez zelenú, modrú a všetky jej odtiene opäť po červenú (360°):



Obrázok 18 - Odtieň [41]

Sýtosť (saturation) zobrazuje stupeň odlišnosti farby od šedej, 0% vyjadruje úplnú absenciu farby (šedá farba), 100% zobrazuje maximálnu intenzitu farby.



Obrázok 19 - Sýtosť [41]

Svetlosť (lightness) / jas (brightness) vyjadruje intenzitu osvetlenia. 0% znamená absenciu svetla, teda čiernu farbu. V modeli HSV hodnota 100% vyjadruje plne saturovanú farbu, v modeli HSL 100% znamená bielu farbu, plnú intenzitu dosiahneme pri použití 50% hodnoty jas.

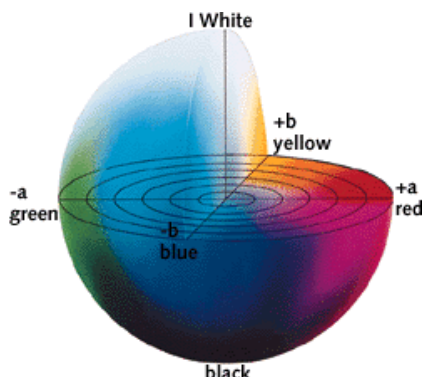
1.2.6.2 Modely farbenia nezávislé na zariadení

CIE

Štandard CIE je zastupovaný medzinárodnou komisiou pre osvetlenie (International Commission on Illumination), ktorá vznikla v roku 1913 za účelom poskytnutia priestoru na výmenu návrhov, nápadov a informácií s cieľom vytvoriť nové metódy pre meranie hodnôt obrazu, vytvoriť modely, ktoré nebudú závislé na zariadeniach. Tieto metódy sú založené na ľudskom vnímaní farieb [2]:

CIEXYZ používa hodnoty X,Y,Z, kde Y vyjadruje jas, X,Z reprezentuje farbu.

CIELAB je založený na vnímaní ľudským okom. Bolo dokázané, že ľudské oko vníma farebnú informáciu ako hodnotu medzi čiernou a bielou, modrou a žltou, červenou a zelenou. Model má 3 parametre – L (jas), parameter a (určuje hodnotu medzi červenou a zelenou) a parameter b (hodnota medzi žltou a modrou). Je reprezentovaný guľou, vertikálna os zobrazuje jas, horizontálne farebné spektrum je definované parametrami a a b . [3]



Obrázok 20 - CIElab [34]

Šedotónový obraz

Pre spracovanie a rozpoznávanie často nepotrebujeme informáciu o farebných zložkách snímky, preto je obraz transformovaný do šedotónového obrazu. Na prevod je použitý nasledujúci vzorec (R – červená farba, G – zelená farba, B – modrá farba):

$$F = 0,299 * R + 0,587 * G + 0,114 * B$$

Rovnica 1 - Výpočet novej hodnoty v šedotónovom modeli

Hodnoty váh boli stanovené na základe citlivosti ľudského oka. [34]

2 Analýza požiadaviek

Cieľom projektu Nemocničný informačný systém – Detekcia nádoru mozgu ako aj predkladanej diplomovej práce je vytvoriť aplikáciu pre potreby lekárov, spracovať, vyhodnotiť a zobraziť výsledky magnetickej rezonancie vo forme grafov, modelov a hodnôt. Doterajšie aplikácie umožňujú výsledky zobrazovať, ale nedokážu ich spracovať a poukázať na existenciu a polohu anomálií.

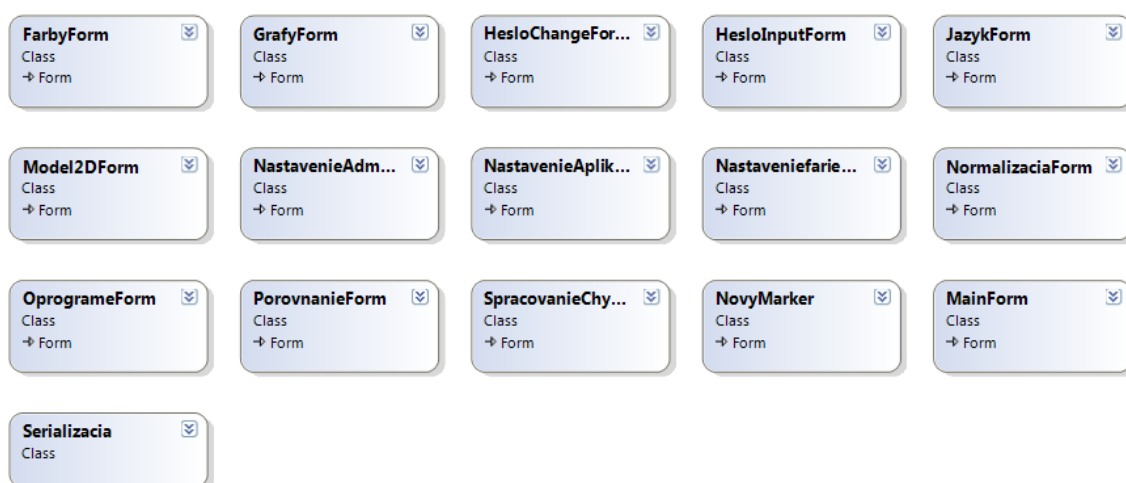
Nevyhnutnou súčasťou spracovania je získané údaje „vyčistiť“, teda odstrániť hodnoty markerov s veľkou chybou merania, resp. nahradiť ich novými hodnotami. Pri spracovaní je dôležité zobraziť kritické miesta – anomálie alebo bunky, ktorých hodnota markera presahuje kritickú hodnotu, pričom kritická hodnota musí byť nastaviteľná používateľom aplikácie. Okrem kritickej hodnoty patrí k ďalším parametrom aj hodnota tolerancie, ktorá je využívaná pri porovnávaní výsledkov u jedného pacienta v čase. Definuje maximálnu prípustnú číselnú hodnotu zmeny výsledkov dvoch alebo viacerých vyšetrení.

Počet rezných rovín závisí od kvality a nastavenia MRI prístroja, preto aj tento parameter musí byť nastaviteľný v aplikácii (navyše, každá os môže mať iný počet rezných rovín). Aplikácia musí taktiež umožňovať porovnanie hodnôt základného markera a aktuálne vybratého markera. Tento marker môže byť buď zo základnej série markerov (26 základných markerov), ktorých hodnota je získaná počas vyšetrenia, alebo to môže byť marker, ktorý je definovaný v aplikácii a vzniká matematickým spracovaním hodnôt základných markerov (napr. $3*Ala-2,5*Cho$, kde *Ala* a *Cho* patria k základným markerom vyhodnocovaným počas vyšetrenia pacienta). Tu ale môže nastať situácia, že výsledné hodnoty môžu byť záporné, resp. mimo intervalu $<0 ; 1>$, preto je potrebné tieto údaje upraviť normalizáciou.

3 Riešenie a implementácia

3.1 Moduly, class diagramy, popis tried

Celý projekt sa skladá z troch základných modulov – *detekcia_nadoru_mozgu*, ktorý obsahuje spracovanie výsledkov magnetickej rezonancie, metódy na prípravu údajov pre potreby zobrazovania, odstraňovanie chybových hodnôt z meraní a taktiež samotné modely vo forme grafov a 2D modelov a taktiež porovnanie výsledkov z rôznych časových období (obr. 21). Z tohto modulu by bolo možné ešte oddeliť časť *Resources*, ktorá obsahuje jazykovú podporu aplikácie (obr.22).

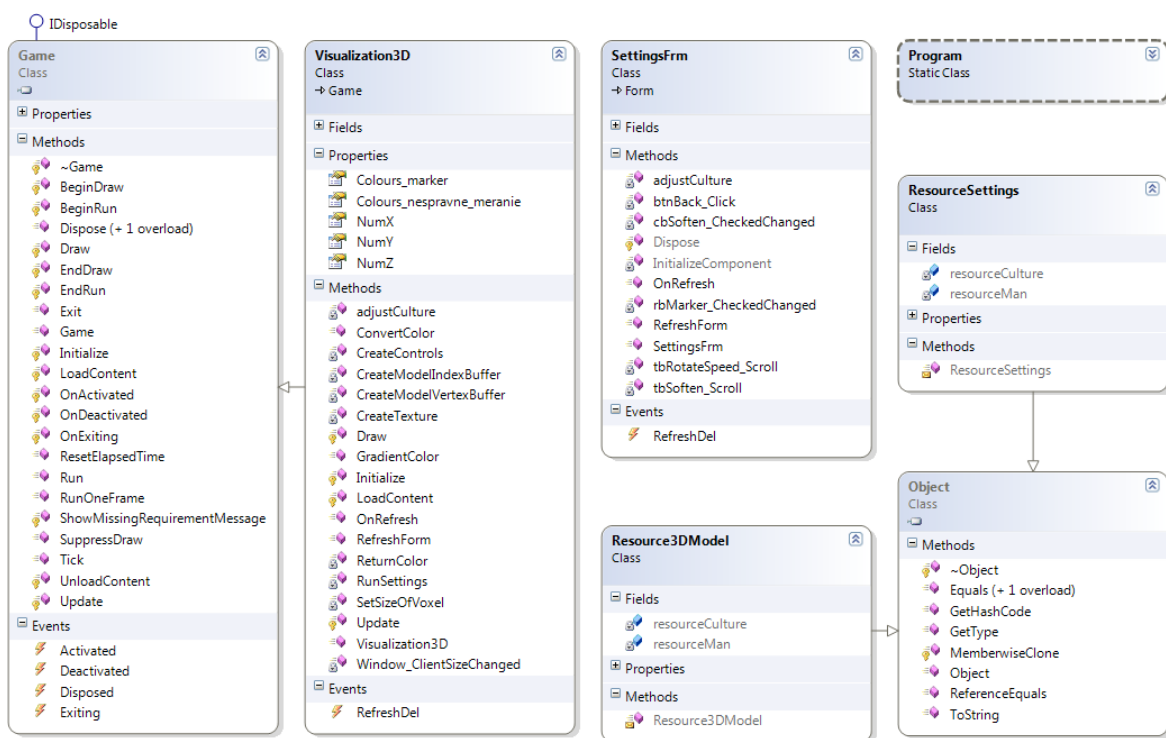


Obrázok 21 - Class diagram projektu *detekcia_nadoru_mozgu*



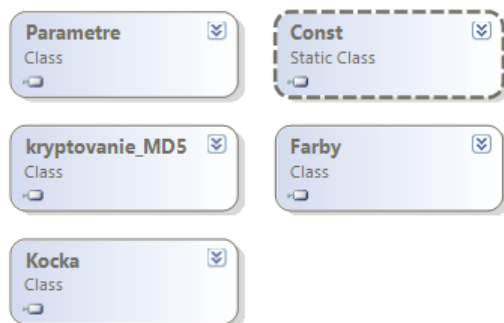
Obrázok 22 - Resources

Druhým projektom sa zaoberá diplomová práca Bc. J. Mešinu - 3D vizualizácia mozgu človeka. Jedná sa o modul pre vytvorenie 3D modelu mozgu, v ktorom sa zobrazujú jednotlivé hodnoty merania (spracované v module *detekcia_nadoru_mozgu*). Každá vrstva 3D modelu reprezentuje rez mozgu v podobe 2D. Samotné rezy sú zobrazované na základe výberu používateľa s cieľom zobraziť anomálie, ktoré môžu znamenať potenciálny nádor alebo iný medicínsky významný nález. Použitá knižnica „XNA Game Studio 4.0 Refresh“ je založená na HW akcelerácii a je často používaná v hrách. Umožňuje integráciu do väčšiny moderných technológií (.NET Framework, Silverlight, WPF, ...). [35] [15]



Obrázok 23 - Class diagram – 3D_vizualizacia

Posledným modulom, ktorý spája projekty *detekcia_nadoru_mozgu* a *3D_vizualizacia* je modul *CommonClasses*, ktorý je súčasťou tejto diplomovej práce a obsahuje triedy potrebné pre komunikáciu s modulom 3D vizualizácia.



Obrázok 24 – Spoločné triedy oboch projektov - CommonClasses

3.1.1 Trieda *Parametre* definuje aktuálne nastavenie aplikácie:

Double critical_value – vyjadruje maximálnu hodnotu markera, ktorá je ešte považovaná za akceptovateľnú a nie je považovaná za prípadnú anomáliu.

Double tolerancia – vyjadruje maximálny akceptovateľný rozdiel hodnôt medzi rôznymi vyšetreniami v čase. Táto hodnota sa využíva pri sledovaní rastu nádoru, resp. zmeny v čase (napríklad porovnanie pred a po operácii).

Double threshold_value – hodnota zobrazuje aktuálne nastavený prah. Parameter na rozdiel od kritickej hodnoty vyjadruje maximálnu hodnotu chyby merania, ktorú ešte používateľ akceptuje. Pokiaľ chce znížiť hodnotu chyby, môže vypočítať novú hodnotu markera a chyby z okolia.

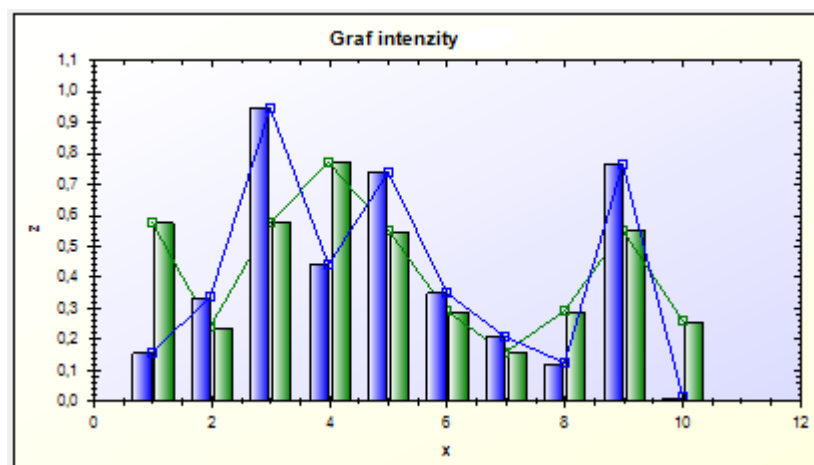
Bool normalizacia_podla_maxima, normalizacia_podla_rozsahu – štandardný prednastavený typ normalizácie, ktorý nemusí byť totožný s aktuálnym spôsobom normalizácie hodnôt zo vstupu. Aktuálny typ normalizácie je uložený v objekte triedy *Kocka* (popísaná neskôr) a zobrazený v spodnej časti hlavného okna.

Kritická hodnota: 0,9	Tolerancia: 0,01	Nastavený marker Ala
------------------------------	-------------------------	-----------------------------

Obrázok 25 - Informácie o aktuálnom nastavení aplikácie

Int pocet_x, pocet_y, pocet_z – atribúty definované rozlišovacou schopnosťou prístroja, pomocou ktorej bolo vyšetrenie realizované - počet meraných bodov v jednotlivých osiach, teda počet možných rezov v jednotlivých osiach. Tieto počty taktiež nemusia byť totožné s rozlišovacou schopnosťou práve zobrazených výsledkov vyšetrenia pacienta.

Color farba_grafu_akt, farba_grafu_cholin – aktuálne nastavená farba pre zobrazenie hodnôt zvoleného markera a základného markera (cholín) v grafoch – čiarový a stĺpcový. Nie je odporúčané nastaviť rovnakú farbu, pretože oba grafy je možné zobrazit' do spoločného grafu s cieľom porovnávania hodnôt v jednotlivých bunkách.



Obrázok 26 - Ukážka grafov v module Grafy

Ostatné atribúty sú špecifikované v programátorskej príručke, ktorá je prílohou tejto práce.

Základné metódy Triedy *Parametre*:

`public string` Najdi_definiciu_markera(`string` nazov_markera) – na základe názvu markeru, ktorý je vstupom funkcie, je vyhľadaná jeho definícia. Pokiaľ nie sú definované nové markery, názov a definícia sú totožné.

`public string` Najdi_definiciu_chyby(`string` nazov_markera) – vstupom funkcie je opäť názov markeru, funkcia však vráti spôsob výpočtu chyby markeru. Pre štandardné markery je táto hodnota definovaná vo vstupnom súbore získanom počas vyšetrenia pacienta (napr. marker *Ala*, jeho chyba je definovaná v stĺpci *Ala_SD*).

Ostatné metódy, ich parametre a použitie sú vysvetlené v programátorskej príručke, ktorá je prílohou tejto práce.

3.1.2 Trieda *Farby*

Trieda *Farby* bola vytvorená s cieľom farebne rozlíšiť hodnoty markerov a chyby merania a zjednodušiť tak zobrazenie výsledkov pre lekára, ktorý posudzuje výsledky. Farebne sa teda oddeľujú anomálie od hodnôt charakterizujúcich zdravé bunky.

Trieda *Farby* obsahuje ako atribúty polia farieb, ktoré sú definované minimom a maximom hodnôt markerov, pre ktoré je farba platná. Farby sú definované ako lineárna zmena z bielej na červenú farbu (hodnoty markerov). Tieto farby si používateľ môže ľubovoľne meniť. Pred potvrdením zmien je volaná funkcia `public bool`

SkontrolujVsetkoZDatagridu(DataGridView dgv), ktorá kontroluje, či sa farebné intervaly medzi sebou neprekrývajú. Taktiež sa kontroluje, či intervaly pokrývajú celý interval hodnôt $<0 ; 1>$. Ak sú všetky podmienky splnené, hodnoty sú potvrdené a zapísané procedúrou `public void UlozVsetko(DataGridView dgv)` do štruktúry.

Metóda `public string VratInterval(double hodnota)` na základe hodnoty markera vráti hranice intervalu, do ktorého hodnota patrí vo forme reťazca. Funkcia `Color VratFarbu(double hodnota, double chyba, double threshold, bool akceptuj_threshold)` obsahuje štyri vstupné parametre, jej návratová hodnota je farba, ktorá sa má zobrazit' v modeloch 2D, resp. 3D. Parameter *hodnota* vyjadruje hodnotu markera v danom zvolenom mieste, parameter *chyba* vyjadruje chybu merania v danom mieste, *threshold* je aktuálna prahová hodnota – maximálna hodnota chyby, ktorú je možné akceptovať. Posledný vstupný parameter vyjadruje, či sa pri hľadaní farby má brať ohľad na hodnotu chyby alebo nie, teda či je v prípade väčšej chyby ako je maximum vrátená farba meranej hodnoty alebo informácia, že hodnota prekročila maximálnu mieru chyby, ktorú akceptujeme a hodnota je teda neznáma:

```
//ak je chyba merania väčšia ako threshold, vráti sa neznáma farba
if (chyba > threshold && akceptuj_threshold) return Const.color_threshold;
```

3.1.3 Trieda Kocka

Atribúty *mozog_akt_marker*, *mozog_akt_cholin* ukladajú hodnoty zvoleného markera a štandardného markera (cholínu) v celom meranom spektre (hlava, mozog). Ku každému z týchto atribútov existuje štruktúra rovnakej veľkosti s údajmi o chybe merania v danom mieste (*mozog_akt_chyba_marker* a *mozog_akt_chyba_cholin*). Načítané hodnoty však často (najmä na okrajoch vyšetrovanej oblasti) nadobúdajú vysokú chybu merania a preto je možné tieto hodnoty nahradiť novými, vypočítanými z okolia – v prípade, že sa hodnota chyby zmenší. Ak je hodnota zmenená, uloží sa táto informácia, aby bolo možné pôvodné a upravené hodnoty v modeloch odlíšiť (*mozog_zmena_marker*, *mozog_zmena_cholin*). Veľkosť a rozlíšenie meranej oblasti sú uložené v atribútoch *kocka_pocet_x*, *kocka_pocet_y*, *kocka_pocet_z* pre jednotlivé osi.

Metódy triedy Kocka

```
public void InicializujMarker(out double[, ] vstup, int x, int y, int z)
public void InicializujChybu(out double[, ] vstup, int x, int y, int z)
public void Inicializuj(out bool[, ] vstup, int x, int y, int z)
```

Veľkosť štruktúry je definovaná parametrami *x*, *y*, *z*, ktorých hodnoty sú zapísané cez „property“ do parametrov *kocka_pocet_x*, *kocka_pocet_y*, *kocka_pocet_z*. Následne sú

štruktúry markerov naplnené hodnotami „-1“, čo zodpovedá nemeranej hodnote, resp. hodnotou „FALSE“, vyjadrujúcou, že informácie neboli spracované. Hodnoty chyby sú nastavené na maximálnu hodnotu (*Const.max_chyba* = 100%).

Funkcia `public void Napln(double[, ,] vstup, DataGridView vstup_dg, int pozicia)` zabezpečuje naplnenie údajov zo vstupu (datagrid), *pozícia* definuje požadovaný spracovávaný stĺpec.

Metódy realizujúce normalizáciu hodnôt:

○ Normalizácia podľa maxima.

```
public void Normalizuj(double[, ,] vstup, double maximum)
if (vstup[i, j, k] != -1) vstup[i, j, k] = (vstup[i, j, k] / maximum);
```

○ Normalizácia podľa rozsahu.

```
public void Normalizuj(double[, ,] vstup, double minimum, double maximum)
if (vstup[i, j, k] != -1) vstup[i, j, k] = ((vstup[i, j, k] - minimum) /
(maximum - minimum));
```

V prípade, že hodnoty markera nadobúdajú aj záporné hodnoty, musí byť pred normalizáciou obor hodnôt posunutý do kladnej osi a teda, ku každej hodnote sa pripočíta hodnota minima celej štruktúry a následne sa normalizuje podľa výberu používateľa (normalizácia podľa rozsahu, normalizácia podľa maxima).

Ak hodnota chyby prekračuje limit (hodnota threshold), je možné pokúsiť sa o výpočet novej hodnoty z okolia, ktorá by lepšie odzrkadľovala realitu (hodnota chyby by sa zmenšila). Princíp definície okolia je vysvetlená v časti implementácia – Spracovanie chybných údajov. Výpočet nových hodnôt zabezpečujú nasledujúce funkcia *VypocitajNovuHodnotu* a *SpracujUdaj*:

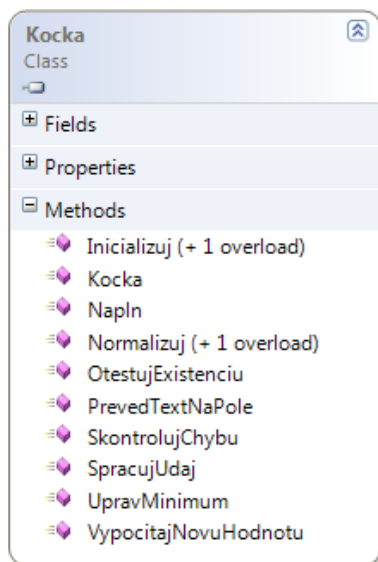
```
public double VypocitajNovuHodnotu(double[, ,] vstup, int param_x, int param_y,
int param_z) – výpočet novej hodnoty z okolia
    for (int i = -1; i < 2; i++)
        for (int j = -1; j < 2; j++)
            for (int k = -1; k < 2; k++)
            {
                if (i == 0 && j==0 && k==0) continue;
                if (OtestujExistenciu(vstup, x+i, y+j, z+k))
                {
                    pocet++;
                    sucet += vstup[x+i, y+j, z+k];
                }
            }
    return (sucet / (double)pocet);
```

Funkcia *VypocitajNovuHodnotu* definuje novú hodnotu ako priemer hodnôt buniek okolia, kde x_{ijk} vyjadruje novodefinovanú hodnotu voxelu na pozícii (i, j, k) . Do novej hodnoty nepočítame súčasnú hodnotu voxelu, pretože ju nepovažujeme za dôveryhodnú.

$$x_{ijk} = \left(\sum_{a=-1}^1 \sum_{b=-1}^1 \sum_{c=-1}^1 x_{(i+a)(j+b)(k+c)} \right) - x_{ijk}$$

Rovnica 2 - Výpočet novej hodnoty voxelu

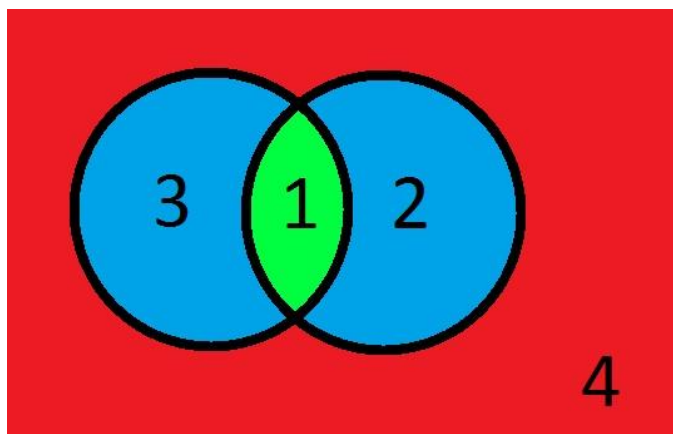
`Public bool SpracujUdaj(double[, ,] vstup_marker, double[, ,] vstup_chyba, int param_x, int param_y, int param_z)` – táto funkcia v tele vypočíta novú hodnotu pomocou vyššie uvedenej funkcie. Ak je nová chyba menšia ako doterajšia, hodnoty markera a chyby sa aktualizujú a hodnoty sa uložia (návratová hodnota bude „TRUE“). V opačnom prípade sa nič nezmení a funkcia vráti hodnotu „FALSE“.



Obrázok 27 - Trieda Kocka

Štandardný vstupný súbor aplikácie je vygenerovaný CSV (Comma Separated Value) súbor počas vyšetrenia pacienta. Ukážkové súbory dobrovoľníkov a testovacie údaje boli poskytnuté Fakultnou nemocnicou v Martine. Každý súbor obsahuje 58 stĺpcov. Prvý stĺpec obsahuje ID riadku, ďalšie tri stĺpce definujú pozíciu meraného prvku v 3D priestore (os x , y , z). Ostatných 27 stĺpcov obsahuje hodnoty markerov (*Ala*, *Cho*, *Gln*, *Glu*, *Ins*, *Lac*, ...) – zoznam všetkých štandardných markerov je uvedený v prílohe. Keďže spracovanie nie je rovnomerné, nie je zaťažené jednotnou chybou, teda pre každý marker a pozíciu je zvlášť uložená hodnota chyby (*Ala_SD*, *Cho_SD*, *Gln_SD*, *Glu_SD*, *Ins_SD*, *Lac_SD*, ...),

čo predstavuje ďalších 27 stĺpcov. Nasledujúci obrázok zobrazuje princíp vzniku rozdielnych chýb merania. Najlepšie zmeraná oblasť je zelená – bola meraná aj modrou oblasťou 2 aj oblasťou 3. Červená oblasť obsahuje najväčšiu hodnotu chyby. Chyby sa hromadia hlavne v okrajových častiach obrazca.



Obrázok 28 - Princíp merania

Niektoré prístroje (napr. MRI prístroj používaný v nemocnici v Martine - Siemens Symphony 1,5 Tesla - ktorá poskytna testovacie údaje) ukladajú aj pomer medzi markerom a hodnotou cholínu v danom mieste, tieto údaje sa však dajú vypočítať (definíciou nového markera) a preto ich nepoužívame. Nasledujúci obrázok ukazuje štruktúru vstupného súboru:

ID	x	y	z	Ala	Ala_SD	Cho	Cho_SD	Cr	Cr_SD	Gln	Gln_SD	
0	0	0	0	0	44,6508	9,8641	84,9545	11,1195	10,3121	35,4200	91,3887	8,5895
1	0	0	0	1	15,1078	32,4787	12,2794	20,9175	12,9528	37,7381	19,7908	4,2968
2	0	0	0	2	33,5510	7,2037	41,5312	11,8363	85,8467	50,9073	3,0299	9,5060
3	0	0	0	3	3,4207	12,3078	33,6234	4,1083	98,1959	11,4451	91,4263	6,8045
4	0	0	0	4	43,9448	33,9521	89,9960	25,2200	72,1933	9,2013	20,0642	1,7917
5	0	0	0	5	69,7599	19,4453	35,5144	2,9355	71,4408	49,0961	41,6109	2,6875
6	0	0	0	6	65,0320	9,5264	66,7199	6,6280	29,8880	25,2115	93,0500	2,9753
7	0	0	0	7	1,4384	16,5044	93,9893	7,2396	1,5230	14,5102	81,7068	12,0204
8	0	0	0	8	20,4445	44,6979	12,2822	32,6038	7,0209	23,2898	32,3844	4,2939
9	0	0	0	9	20,0698	35,1697	42,3146	37,8528	42,4324	43,6387	65,4374	6,7000
10	0	1	0	0	12,8794	24,9478	60,4904	18,3295	65,9901	5,8907	93,4894	10,3847
11	0	1	1	1	56,7193	36,6331	94,6579	27,8126	3,0722	11,9777	79,4823	8,9108
12	0	1	2	2	22,2727	49,8133	61,2960	14,1833	4,2136	1,7670	0,1328	4,2313
13	0	1	3	3	21,5970	23,6323	26,2960	7,7906	73,7749	41,1559	91,4682	18,2253
14	0	1	4	4	48,6578	12,4129	69,7851	6,2195	85,7321	43,0987	52,7411	14,4860
15	0	1	5	5	56,6798	49,8944	14,6887	18,9996	50,4237	32,3032	0,6712	20,0317
16	0	1	6	6	33,5048	14,3867	70,2553	33,8615	25,8579	24,5258	3,7686	10,4622
17	0	1	7	7	22,2317	27,0584	67,9899	1,6282	45,5352	13,2725	13,6548	9,1062
18	0	1	8	8	17,0507	40,1122	13,8136	12,7617	42,3773	15,2529	99,5062	4,5308
19	0	1	9	9	42,8158	22,2382	35,7041	4,2175	98,3696	30,7068	10,3169	1,6205
20	0	2	0	0	20,9364	7,6711	14,1621	16,7781	65,6750	3,3562	1,3261	8,7574
21	0	2	1	1	33,6249	12,7817	97,2595	24,2714	82,9292	47,1215	57,9964	1,1039
22	0	2	2	2	62,3389	37,6146	2,4034	23,1118	43,4050	43,2558	98,5608	19,8004
23	0	2	3	3	69,2409	14,4363	33,0675	35,2960	29,2899	28,0929	81,6999	7,8206

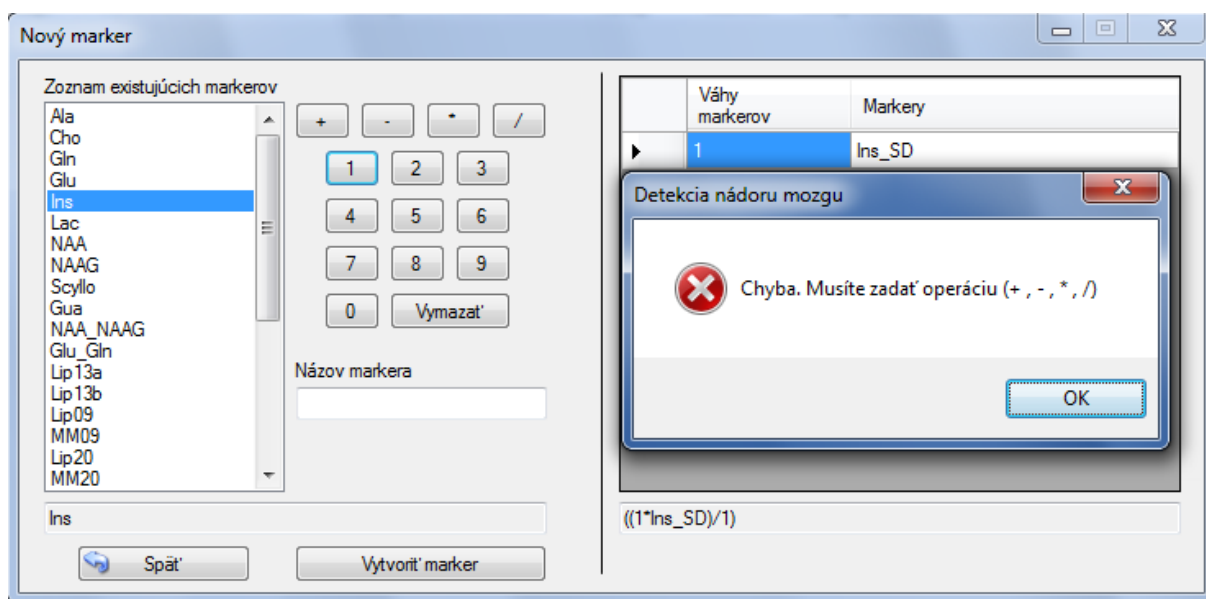
Obrázok 29 - Štruktúra vstupného súboru

Aplikácia pracuje naraz vždy s dvomi markermi, pričom jeden z nich je fixne definovaný základný marker – cholín. Druhý marker je vybraný používateľom. Všetky

základné markery, ako aj používateľom vytvorené markery môžu byť ďalej matematicky spracované a použité na tvorbu ďalších markerov. Na tento účel je vytvorený modul *Nový Marker*, v ktorom možno definovať nové markery. V ľavej časti okna sa definuje spôsob výpočtu nového markera. O správnosť definície sa starajú tri funkcie:

- `bool SkontrolujSpravnost_Marker()`
- `bool SkontrolujSpravnost_Cislo(int cislo)`
- `bool SkontrolujSpravnost_Operacia()`

Ich úlohou je zabezpečiť, aby novo definovaný marker mohol byť vypočítaný korektne. Teda po vybraní operácie (+, -, *, /) musí nasledovať buď číslca alebo názov markera a pod.



Obrázok 30 - Tvorba nového markera

Na obrázku môžeme vidieť vybraný marker „*Ins*“ pre definíciu. Následne bola vybratá číslca 1, ktorej vloženie by bolo ale nekorektné a preto je používateľ upozornený a požiadaný o opravu.

To však ešte pre uloženie markera nestačí, je potrebné zabezpečiť, aby novovytváraná definícia obsahovala aspoň 1 marker a nebola to teda konštanta. Taktiež definícia nemôže končiť operáciou:

```
//definícia markera nesmie končiť operáciou (napr. 3*Ala+5+)
if (!SkontrolujSpravnost_Operacia())
{
    string text = rm.GetString("Chyba_nekorektne_ukoncenie", culture);
    MessageBox.Show(text, nazov_programu, MessageBoxButtons.OK,
        MessageBoxIcon.Error);
}
```



```

        return;
    }
    //nebol použitý žiadny marker, definícia je len konštanta
    if (pocet_pouzitych_markerov == 0)
    {
        string text = rm.GetString("Chyba_len_konstanty", culture);
        MessageBox.Show(text, nazov_programu, MessageBoxButtons.OK,
            MessageBoxIcon.Error);
        return;
    }

```

Na ukladanie novo definovaných markerov slúžia 3 zoznamy reťazcov:

```

//zoznam všetkých markerov, ktoré môžu byť použité.
public List<string> zoznam_markerov = new List<string>();
//definícia markera na výpočet hodnoty markera
public List<string> definicia_markerov = new List<string>();
//definícia spôsobu výpočtu chyby
public List<string> definicia_chyb = new List<string>();

```

Všetky zoznamy obsahujú nielen vytvorené markery, ale aj definície štandardných markerov, ktorých hodnoty sú získané priamo počas vyšetrenia a netreba ich ďalej v aplikácii vypočítavať. Tieto zoznamy sú definované v triede „Parametre“ a ich hodnoty sú pri ukončení programu uložené a po spustení opätovne načítané, teda nastavenia programu zostávajú totožné aj po ďalšom spustení aplikácie.

Nová hodnota markera, ktorý vznikol z iných markerov, je vyjadrená v rovnici č.3 – množina M obsahuje všetky použité markery v definícii (teda nielen štandardné, ale aj vytvorené aplikáciou, definícia však musí obsahovať aspoň 1 marker), w_m je váha markera v definícii, c – konštanta, pripočítaná k definícii (definícia túto konštantu samozrejme obsahovať nemusí).

$$(\sum_{m \in M} w_m * m) + c = \text{novy_marker}$$

Rovnica 3 - Výpočet hodnôt nových markerov

Hodnota chyby je vypočítaná váženým priemerom použitých markerov:

$$\frac{(\sum_{m \in M} w_m * m)}{|M|} = \text{chyba}$$

Rovnica 4 - Výpočet chyby merania

Hodnoty zvoleného markera sa ukladajú ako atribúty v objekte triedy „Kocka“.

```
//hodnoty aktuálne zvoleného markera
double[, ,] mozog_akt_marker = null;
//hodnoty chýb aktuálne zvoleného markera
public double[, ,] mozog_akt_chyba_marker = null;
//hodnoty cholínu - cholín sa považuje za defaultný marker
public double[, ,] mozog_akt_cholin = null;
//hodnoty chyb cholínu
public double[, ,] mozog_akt_chyba_cholin = null;
```

Prístup k súboru je realizovaný pomocou databázových príkazov. Pred samotným spustením požiadavky je potrebné zistiť definíciu markera, nie len jeho názov, pretože sa môže jednať o marker, ktorý nepatrí medzi štandardné a je vytvorený používateľom a teda sa jeho hodnoty v súbore priamo nenachádzajú. [12] [13]

```
String definicia_markera = parametre.Najdi_definiciu_markera(parametre.Marker);

string definicia_markera_chyba =
parametre.Najdi_definiciu_chyby(parametre.Marker);

OleDbCommand command = new OleDbCommand("SELECT x, y, z," + definicia_markera +
"," + definicia_markera_chyba + "," + "Cho" + "," + "Cho_SD" + " FROM [ExcelData$]",
connection);
```

Údaje načítané zo súboru však nemusia mať požadovaný tvar. Je to spôsobené hlavne definovaním vlastných markerov, kde hodnota môže nadobúdať aj zápornú hodnotu. Preto je celý obor hodnôt posunutý do kladnej osi a hodnoty sú následne normalizované do intervalu $<0; 1>$. Aplikácia umožňuje 2 režimy normalizácie:

- Normalizácia podľa maxima, kde sa každá hodnota vydolí maximom:
 $(vstup[i, j, k] / maximum)$
- Normalizácia podľa rozsahu – nová hodnota vznikne po odpočítaní maxima od hodnoty pôvodného vstupu a následnom vydelení rozsahom:
 $((vstup[i, j, k] - minimum) / (maximum - minimum))$

Hodnoty minima a maxima sú tiež získané pomocou databázových príkazov [11]:

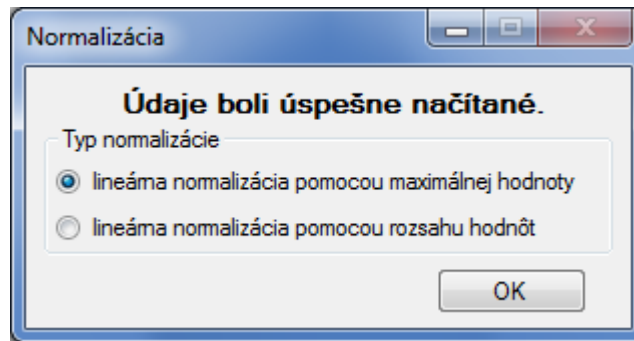
```
//zistenie maxima
command = new OleDbCommand("SELECT max(" + "Cho" + ") FROM [ExcelData$]",
connection);
maximum = (double)command.ExecuteScalar();
//zistenie minima
command = new OleDbCommand("SELECT min(" + "Cho" + ") FROM [ExcelData$]",
connection);
minimum = (double)command.ExecuteScalar();
Ak hodnota minima je záporná, je potrebné hodnoty posunúť do kladnej časti.
if (minimum < 0)
```

```
{ kocka.UpravMinimum(kocka.mozog_akt_cholin, minimum);
  maximum += Math.Abs(minimum);}
```

Na úpravu hodnôt slúži procedúra UpravMinimum definovaná v triede „Kocka“:

```
for (int i = 0; i < x; i++)
    for (int j = 0; j < y; j++)
        for (int k = 0; k < z; k++)
            if (vstup[i, j, k] != -1)
                vstup[i, j, k] += minimum;
```

Následne je možné hodnoty normalizovať podľa typu, ktorý si vyberie používateľ:

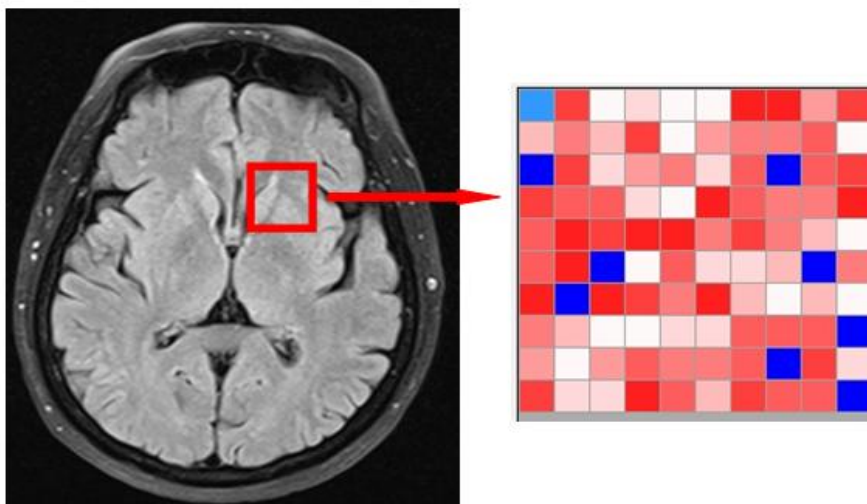


Obrázok 31 - Typ normalizácie

Akonáhle používateľ zvolený marker zmení, údaje sa zo súboru načítajú opäť. Teda v pamäti je naraz uložený len aktuálne vybraný marker a hodnota jeho chyby.

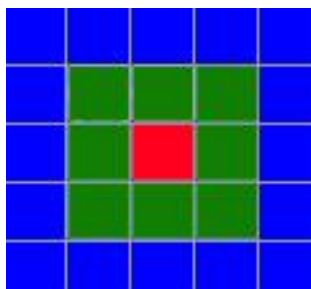
3.2 Spracovanie údajov s vysokou hodnotou chyby

Údaje, ktoré sú získané vyšetrením pacienta magnetickou rezonanciou, nemusia mať v každom bode (voxel) požadovanú kvalitu. V niektorých miestach môžu nadobúdať nedefinované, resp. nesprávne vyhodnotené hodnoty, ktoré by mohli naznačovať rizikovú oblasť (nádor). Preto je potrebné tieto údaje „vyčistiť“ a odstrániť hodnoty, ktorých chyba je príliš vysoká. Používateľ si nastavuje prahovú hodnotu (threshold), ktorá vyjadruje maximálnu hodnotu chyby (v percentách), ktorú je ochotný ešte akceptovať. Všetky hodnoty s chybou väčšou ako tento prah, sú považované za nedôveryhodné (označené modrou farbou, obr.32).



Obrázok 32 - 2D model - rez mozgu

Ak ale lekár potrebuje zistiť hodnotu aj v týchto miestach, naša aplikácia mu ponúka možnosť výpočtu nových hodnôt týchto prvkov z ich okolia. Okolím šírky 1 „červeného“ prvku rozumieme množinu prvkov, ktoré s ním majú aspoň jednu spoločnú hranu v 3D priestore:



Obrázok 33 - Okolie

Každý prvok má teda najviac 26 prvkov okolia so šírkou 1 (krajné prvky ich majú menej). Nová hodnota prvku vznikne sčítaním hodnôt všetkých existujúcich prvkov okolia a vydelením počtom týchto prvkov. Novú hodnotu prvku však považujeme za korektnú len v prípade, že hodnota chyby sa zníži (hodnota chyby sa vypočíta rovnakým spôsobom ako nová hodnota prvku). Tu je však dôležité zdôrazniť, že nie je nutné, aby sa hodnota chyby znížila pod prahovú hodnotu. Hlavné je, že hodnota chyby sa zmenšila. Nasledujúci vzorec vyjadruje spôsob výpočtu novej hodnoty voxelu z okolia šírky 1, množiny A, B, C definujú prvky okolia v jednotlivých osiach.

$$x_{ijk} = \frac{(\sum_{a=-1}^1 \sum_{b=-1}^1 \sum_{c=-1}^1 x_{(i+a)(j+b)(k+c)}) - x_{ijk}}{|A| * |B| * |C| - 1} \quad \exists x_{(i+a)(j+b)(k+c)}$$

Rovnica 5 - Výpočet novej hodnoty voxelu

V prípade, že novú hodnotu chceme definovať širším okolím, do vzťahu ešte zavedieme váhu jednotlivých hodnôt prvkov okolia (čím bližšie je prvok okolia vybratému voxelu, tým má väčšiu váhu). $w_{(i+a)(j+b)(k+c)}$ vyjadruje váhu prvku, $x_{(i+a)(j+b)(k+c)}$ hodnotu prvku okolia na pozícii $(i+a)(j+b)(k+c)$. Váhu prvku x_{ijk} odporúčame nastaviť na hodnotu „0“, resp. veľmi malú, pretože ju považujeme za nedôveryhodnú a nemala by ovplyvňovať novú hodnotu.

$$x_{ijk} = \frac{\sum_{a \in A} \sum_{b \in B} \sum_{c \in C} x_{(i+a)(j+b)(k+c)} * w_{(i+a)(j+b)(k+c)}}{|A| * |B| * |C|} \quad \exists x_{(i+a)(j+b)(k+c)}$$

Rovnica 6 - Všeobecná rovnica pre výpočet novej hodnoty z okolia ľubovoľnej šírky

Prvky na spracovanie môžeme vyberať ručne z tabuľky (datagrid) alebo môžeme požadovať spracovať všetky prvky podľa zvoleného kritéria, ktorých hodnota chyby je vyššia ako naša tolerancia (threshold). Nasledujúci obrázok zobrazuje informácie o nespracovanom prvku – jeho pozícia (súradnice x, y, z), aktuálna hodnota markera na danej pozícii, jeho farebné rozlíšenie, hodnota chyby s príslušnou farebnou kategóriou:

x	y	z	Hodnota		Hodnota chyby	
1	1	2	0,265664863129673		0,418407	
1	1	3	0,951614613108124		0,483018	
1	1	4	0,761084024480018		0,487844	
1	1	5	0,204042749870456		0,585665	

Obrázok 34 - Voxely pred spracovaním

Po úspešnom spracovaní (úspešným spracovaním sa rozumie zníženie hodnoty chyby) je zobrazená pozícia, hodnota markera a chyby pred a po spracovaní spolu s farebnou príslušnosťou do jednotlivých kategórií:

x	y	z	Hodnota pred spracovaním	Hodnota po spracovaní	Hodnota chyby pred spracovaním	Hodnota chyby po spracovaní
9	5	5	0.326960021321434	0.496263649432078	0.962265	0.388395009858515
9	5	3	0.693520245949473	0.552961253882597	0.791207	0.48193401725449
9	5	2	0.0109629024891188	0.562358672714714	0.70187	0.469790090652372
9	4	10	0.965839140088357	0.551680889048059	0.451115	0.439077702120169
9	4	8	0.551480066144001	0.501762462263061	0.934171	0.464124208073091

Obrázok 35 - Voxely po spracovaní

Na nasledujúcej snímke je zobrazený celý formulár pre spracovanie chybných údajov z merania, prvky, ktoré ešte nie sú spracované a taktiež prvky, ktorých chyba už bola znížená. V pravej časti je navigačný panel pre výber zobrazovaných údajov a nastavenie prahovej hodnoty (threshold).

Obrázok 36 - Formulár - Spracovanie chýb

Novovypočítané hodnoty sú uložené, nahrádzajú hodnotu získanú počas vyšetrenia. Ak je však potrebné zobrazenie pôvodných hodnôt, teda bez spracovania, v hlavnom okne aplikácie je táto možnosť dostupná. Ide vlastne o opätovné načítanie výsledkov vyšetrenia podľa zvoleného markera.

3.3 Nastavenie aplikácie

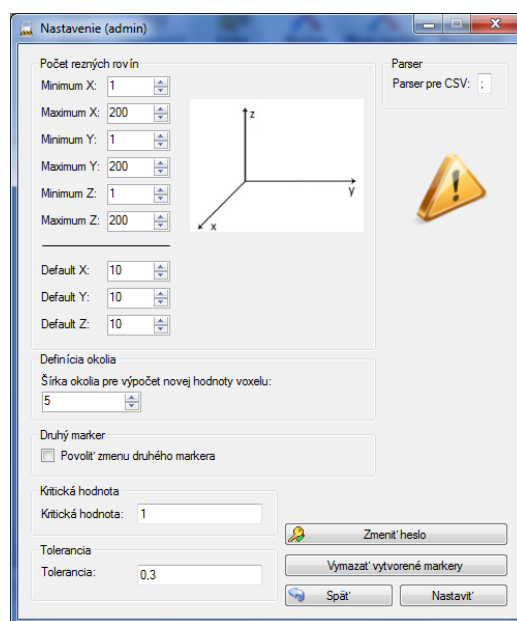
Medzi základné požiadavky aplikácie patrí univerzálnosť, teda naša aplikácia sa nesústreďuje len na jeden konkrétny prístroj, ale ponúka možnosť nastavenia parametrov podľa typu, presnosti a kvality prístroja použitého na vyšetrenie. Konkrétne sa jedná o nastavenie počtu rezných rovín vo všetkých osiach. Základné rozmedzie počtu rezných rovín je $\langle 5 ; 200 \rangle$ s možnosťou zmeny v „admin režime“ (popísaný neskôr). Ďalej je to kritická hodnota, ktorá vyjadruje maximálnu prípustnú hodnotu markera. Akonáhle je hodnota vyššia, používateľ (lekár) je na danú skutočnosť upozornený. Hodnota tolerancie je potrebná pre spracovávanie a porovnávanie výsledkov pacienta z rôznych časových období. Sleduje sa tak vývoj anomálií v čase. Hodnota tolerancie definuje maximálnu hodnotu zmeny v čase. Obe nastavené hodnoty sú zobrazené aj v základnom okne aplikácie.

Používateľ si taktiež môže nastaviť presnosť hodnôt (počet desatinných miest), ako aj predvolené nastavenie typu normalizácie, prahovú hodnotu a farby grafov. Aplikácia ponúka 3 jazykové mutácie – slovenský, anglický a nemecký jazyk.

Obrázok 37 - Formulár - Nastavenie aplikácie

3.4 Nastavenie aplikácie v režime „Admin“

Režim „Admin“ pre nastavenie aplikácie je určený správcovi aplikácie v konkrétnej nemocnici. Nastavuje limitné hodnoty pre počet rezov, predvolenú kritickú hodnotu, hodnotu tolerancie a oddeľovač hodnôt v CSV súbore (štandardný oddeľovač – parser je „;“). Správca systému môže vymazať používateľmi definované nové markery. Jeho režim je kvôli bezpečnosti chránený heslom, ktoré je šifrované algoritmom MD5. V systéme je uložený len odtlačok hesla, správca si ho po prihlásení môže kedykoľvek zmeniť. Z dôvodu bezpečnosti je potrebné zadať aktuálne heslo a dvakrát nové heslo.



Obrázok 38 - Nastavenie aplikácie administrátorom systému

3.4.1 Šifrovací algoritmus MD5

MD5 (Message Digest 5) patrí k známym a rozšíreným kryptografickým algoritmom. Jedná sa o jednocestnú hashovaciu funkciu (teda je k nej náročné nájsť inverznú funkciu), ktorej výsledkom je 128 bitový (16 bytov) hash. Algoritmus bol vytvorený v roku 1991 Ronom Rivestom s cieľom nahradiť nie celkom bezpečnú predchádzajúcu verziu – MD4.

Medzi hlavné požiadavky hashovacej funkcie patrí [42]:

- Pre každú správu je jednoduché vypočítať jej hash.
- Z hodnoty hashu nie je možné určiť pôvodnú správu.
- Je náročné nájsť 2 rôzne správy, ktorých výstup funkcie je totožný (odolnosť voči kolíziám – collision resistance).

Do roku 1995 bol považovaný za úplne bezpečný, v tomto roku sa začali objavovať pochybnosti o úplnej bezkolíznosti (po zadaní dvoch rôznych vstupov algoritmu nikdy nedostaneme rovnaký výstup), ktoré boli potvrdené o rok neskôr. M. Stevens vo svojej knihe *On Collisions for MD5* hovorí, že pri poznaní pôvodnej správy trvá len niekoľko sekúnd nájsť kolíziu (v prípade použitia počítača s procesorom Pentium4 ; 2,6 GHz). [22]

3.4.1.1 Narodeninový útok

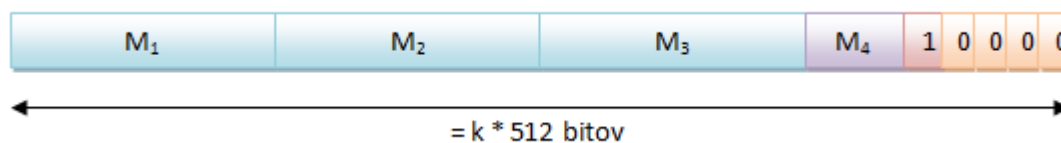
Narodeninový útok je útok zameraný na hashovacie funkcie s cieľom nájsť kolízie. Postup je nasledovný:

- Zvolíme niekoľko správ.
- Vypočítame ich hash.
- Zisťujeme kolízie medzi odtlačkami (hash).

Pri predpoklade zložitosti výpočtu odtlačku $O(1)$, je zložitosť výpočtu odtlačkov $O(k)$, kde k vyjadruje mohutnosť množiny správ. Množina výstupov hashovacej funkcie je konečná (128 bitov), môžeme tieto výsledky triediť lineárnou zložitou. Celková zložitosť narodeninového útoku je teda $O(k)$. Otázkou však je, ako zvoliť hodnotu parametra k , teda počet vzoriek. Čím väčšia je mohutnosť množiny vstupných správ, tým je väčšia pravdepodobnosť objavenia kolízie. Zvyšuje sa tým však nielen časová, ale aj pamäťová náročnosť [43].

3.4.1.2 Algoritmus

Samotný kryptovací algoritmus konvertuje správu ľubovoľnej dĺžky na sekvenciu 128 bitov. Pôvodná správa je rozdelená na bloky o veľkosti 512 bitov. V prípade, že dĺžka správy nie je touto hodnotou deliteľná, doplní sa za správu hodnota 1 a ostatné budú nuly.



Obrázok 39 - Štruktúra správy

Spracovanie správy na rozdiel od algoritmu MD4 pozostáva až zo štyroch kôl. Príkladmi funkcií sú:

$$F(X, Y, Z) = (X \wedge Y) \vee (\neg X \wedge Z)$$

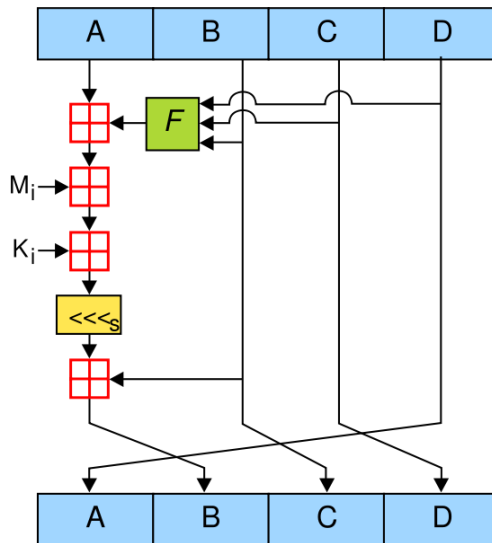
$$G(X, Y, Z) = (X \wedge Z) \vee (Y \wedge \neg Z)$$

$$H(X, Y, Z) = X \oplus Y \oplus Z$$

$$I(X, Y, Z) = Y \oplus (X \vee \neg Z)$$

Rovnica 7 - Funkcie pre šifrovací algoritmus

Operácie $\oplus$, $\wedge$, $\vee$, $\neg$ vyjadrujú XOR, AND, OR alebo NOT. [42]



Obrázok 40 - Šifrovací algoritmus MD5 [42]

3.5 Zobrazovacie módy aplikácie

Každý voxel je charakterizovaný svojou pozíciou v trojdimenzionálnom priestore, hodnotou markera a hodnotou chyby, ktorá vznikla počas merania. Po načítaní výsledkov magnetickej rezonancie je lekár informovaný o parametroch a lokalite nález, resp. anomálie. V prípade, že sa anomália nenašla, zobrazí symbol (obr.41), charakterizujúci zdravie pacienta (tumor nebol nájdený). Anomálie sú charakteristické zvýšenou hodnotou markera v danom mieste. Aplikácia pracuje s 26-timi základnými markermi, ktoré pokrývajú detekciu širokého spektra nádorov. Počas procesu vyšetrenia magnetickej rezonanciou nebudú získavané hodnoty o všetkých markeroch, diagnostika sa špecializuje na predpokladané typy nádorov. Pri spracovaní týchto údajov treba dávať pozor na hodnotu chyby, pokiaľ je vysoká, je možné, že hodnota sa zmerala nepresne a žiadny nádor sa v mieste nenachádza. Všeobecne teda môžeme rozdeliť hodnotu každého voxelu do troch skupín:

- Hodnota voxelu je menšia ako kritická hodnota a aj chyba je menšia ako prah, to znamená, že v danom mieste sa anomália nenachádza. Túto oblasť teda môžeme z ďalšieho skúmania vylúčiť, pretože nepredstavuje riziko nádoru.
- Hodnota voxelu je väčšia ako kritická hodnota a chyba merania je menšia ako prah. Táto situácia hovorí o nájdení miesta, ktorá vykazuje známky aktivity a predstavuje potenciálne nebezpečenstvo. V týchto miestach sa nachádzajú anomálie, na ktoré je potrebné upozorniť lekára.
- Chyba merania je väčšia ako prah. V tomto prípade sme objavili voxel, ktorého hodnota chyby prekračuje povolený limit daný prahovou hodnotou. Tieto voxely sú potenciálne rizikové, je pri nich odporúčané ďalšie spracovanie s cieľom znížiť hodnotu chyby a vypočítať presnejšiu hodnotu markera v danom mieste a teda dokázať rozhodnúť o anomálii, resp. potrebe ďalšieho vyšetrenia (ak chybu nie je možné znížiť pod akceptovateľnú hranicu definovanú parametrom prah (threshold)).



Obrázok 41 - Rizikové bunky neboli identifikované

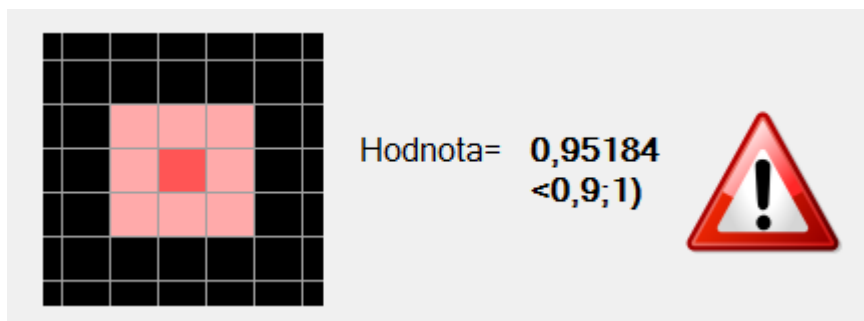
Niektoré MRI prístroje v prípade, že dané miesto nemerali, zapisujú maximálnu hodnotu chyby a markera, iné výsledky nemeraných miest neuvádzajú vôbec. Nasledujúci obrázok zobrazuje centrum nádoru vo voxelu s indexami $x=5$; $y=5$; $z=5$. Okolie tohto centra má tiež zvýšené hodnoty meraného markera (priemerná hodnota v okolí je 0,75; kritická hodnota bola nastavená na 0,7).

	x	y	z	Marker	Chyba merania
	4	6	6	0.772705	0.292752
	5	4	4	0.784589	0.292734
	5	4	5	0.736818	0.240952
	5	4	6	0.746794	0.248074
	5	5	4	0.763963	0.317622
	5	5	5	0.95184	0.311729
	5	5	6	0.75184	0.246009
	5	6	4	0.75184	0.332318
	5	6	5	0.75184	0.246398
	5	6	6	0.75184	0.244928



Obrázok 42 - Hodnoty voxelov, ktoré prekračujú kritickú hodnotu

Nasledujúci obrázok zobrazuje rez osou z ($z=5$). Červený štvorec zobrazuje polohu stredu nádoru, bledočervené štvorce reprezentujú okolie nádoru. Zvyšná čierna časť nebola meraná.

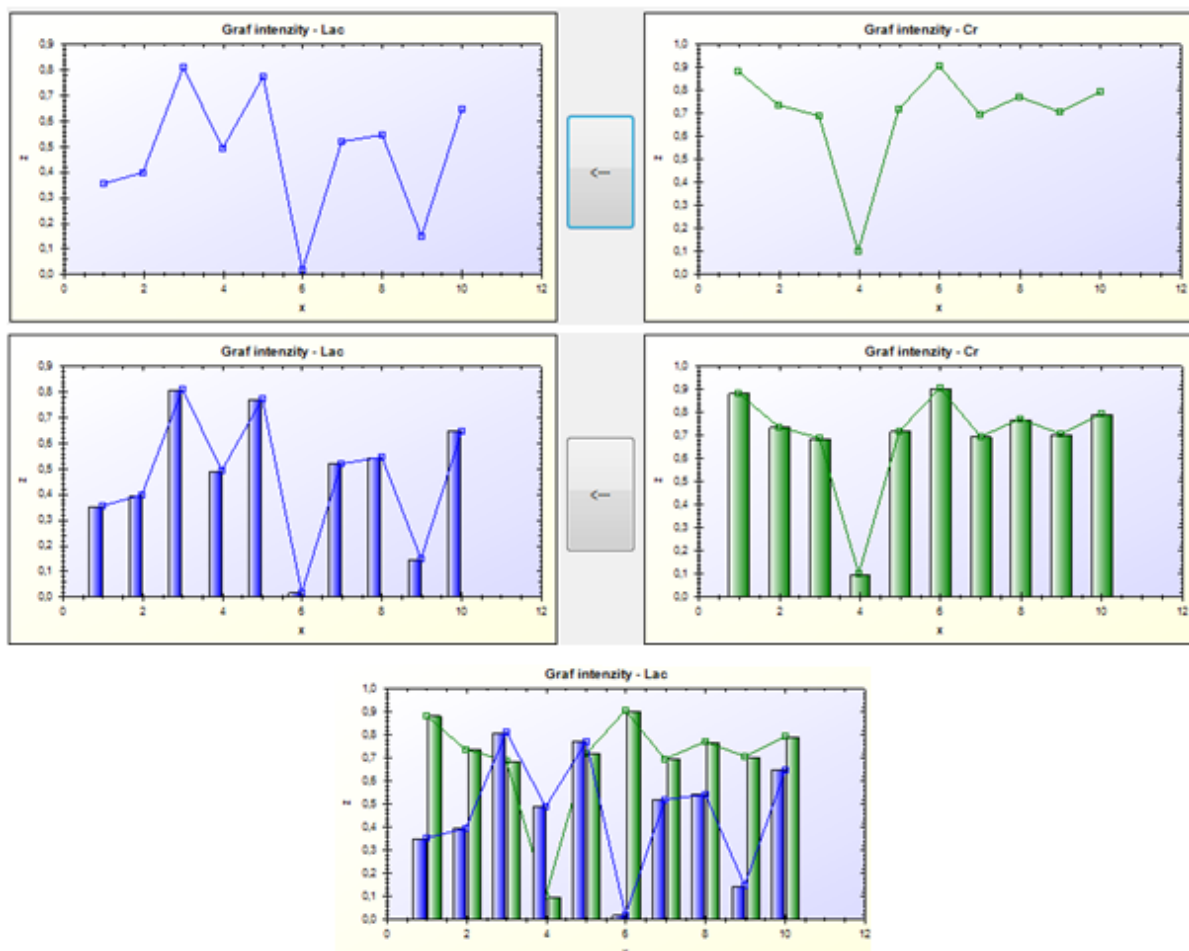


Obrázok 43 - Nádor a jeho okolie

3.5.1 Grafy

Modul Grafy po zvolení rezu dvoch osí vykresľuje hodnoty v tretej súradnici vo forme grafu. Zobrazujú sa hodnoty aktuálne nastaveného markera a cholínu (štandardný marker) s hodnotou chyby menšou ako zvolený prah (threshold). Princíp zníženia chyby merania a výpočet náhradnej hodnoty je popísaný v časti „Spracovanie údajov s vysokou hodnotou chyby“.

Častou požiadavkou je porovnať hodnoty týchto dvoch markerov v každom mieste. Je preto možné oba grafy vykresliť spolu a sledovať tak hodnoty oboch v jednom mieste. Pre potreby grafov je použitá knižnica ZedGraph a využívame čiarový a stĺpcový graf (typ grafu, resp. zobrazovanie oboch si nastavuje používateľ). Taktiež je možné nastavenie farieb grafov pre jednoduchšie porovnávanie.



Obrázok 44 - Grafy

3.5.2 2D modely

Samotné zobrazovanie rezov vo forme 2D modelov obsahuje štyri základné časti:

- zobrazenie hodnôt vybraného markera vo zvolenom reze,
- zobrazenie chyby merania zvoleného markera,
- zobrazenie hodnoty markera cholín,
- zobrazenie chyby merania cholínu.

Všetky tieto hodnoty sú vykreslené v osobitných datagridoch a farebne rozlíšené.

Miera farebnej odlišnosti jednotlivých intervalov je nastaviteľná v module farba, kde sa zvolí počet intervalov, hranice a farba reprezentujúca zvolený interval. V základnom nastavení sa jedná o farebné spektrum bielej a červenej farby (obr. 45) pre hodnoty markera a spektrum zelenej a červenej pre chyby merania vzniknuté počas vyšetrenia MRI prístrojom (obr. 46). Nastavenie farieb je uložené v objekte triedy *Farby*, jeden objekt reprezentuje farebné spektrum pre aktuálny marker a druhý reprezentuje farebné rozlíšenie chýb merania.

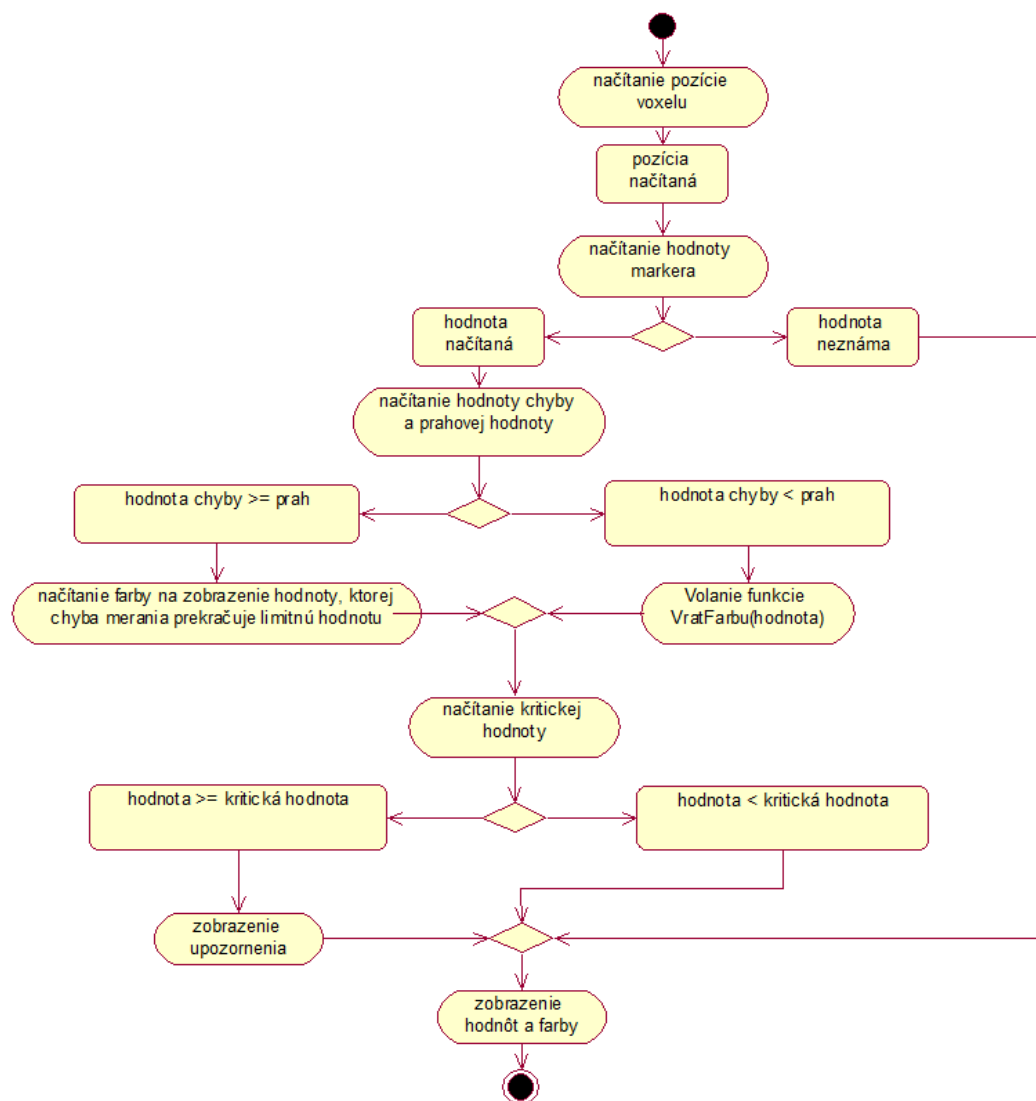
Minimum	Maximum	Farba
0	0,125	
0,125	0,25	
0,25	0,375	
0,375	0,5	
0,5	0,625	
0,625	0,75	
0,75	0,875	
0,875	1	

Obrázok 45 - Nastavené farebné spektrum pre hodnoty markera

Minimum	Maximum	Farba
0	0,01	
0,01	0,05	
0,05	0,1	
0,1	0,2	
0,2	0,3	
0,3	1	

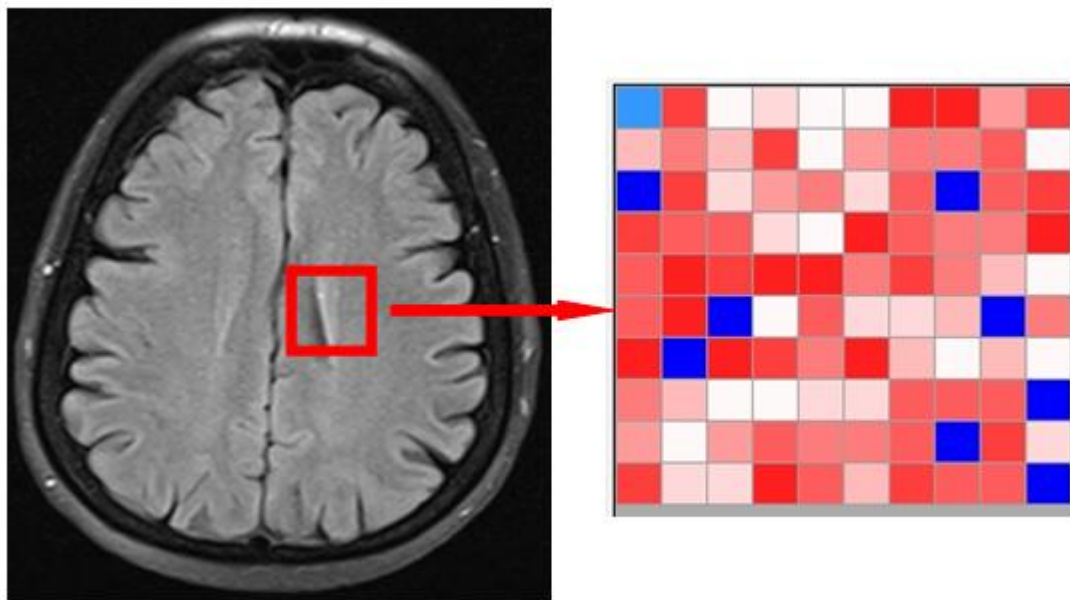
Obrázok 46 - Nastavené farebné spektrum pre chyby merania

O zaradenie hodnoty do správnej skupiny a zobrazenie sa stará funkcia `public Color VratFarbu(double hodnota, double chyba, double threshold, bool akceptuj_threshold)`, ktorej vstupom je hodnota zvoleného markera, chyba merania v danom mieste, zvolená prahová hodnota. V prípade, že chceme zobrazit' farebnú hodnotu bez ohľadu na hodnotu prahu, parameter *akceptuj_threshold* nadobudne hodnotu „FALSE“. Ak pošleme parameter s hodnotou „TRUE“, všetky hodnoty, ktorých chyba presahuje akceptovateľnú hodnotu chyby, sa zobrazia špeciálnou farbou (v našom prípade modrou, táto konštanta je definovaná v triede *Const* pod názvom *color_threshold*). Špeciálnu funkciu v zobrazení má aj čierna farba (definovaná konštantou *neznama_farba* v statickej triede *Const*) vyjadrujúca, že marker v danom mieste meraný nebol.



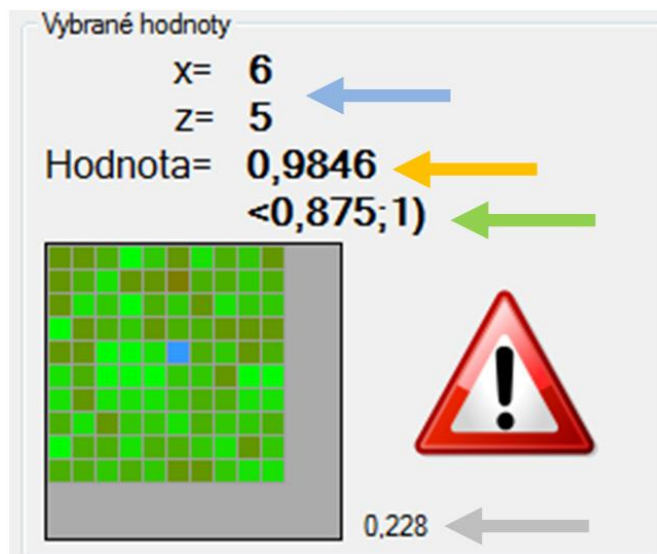
Obrázok 47 - Načítanie hodnôt (diagram)

V prípade, že je potrebná hodnota aj v nemeranej oblasti, resp. presnejšia hodnota s menšou chybou, je možné využiť modul spracovanie chýb a pokúsiť sa chybu merania znížiť a zvýšiť tak dôveryhodnosť hodnoty. Nová hodnota je vypočítaná ako lineárna aproximácia okolia. Postup je bližšie popísaný v časti *Spracovanie údajov s vysokou hodnotou chyby*.



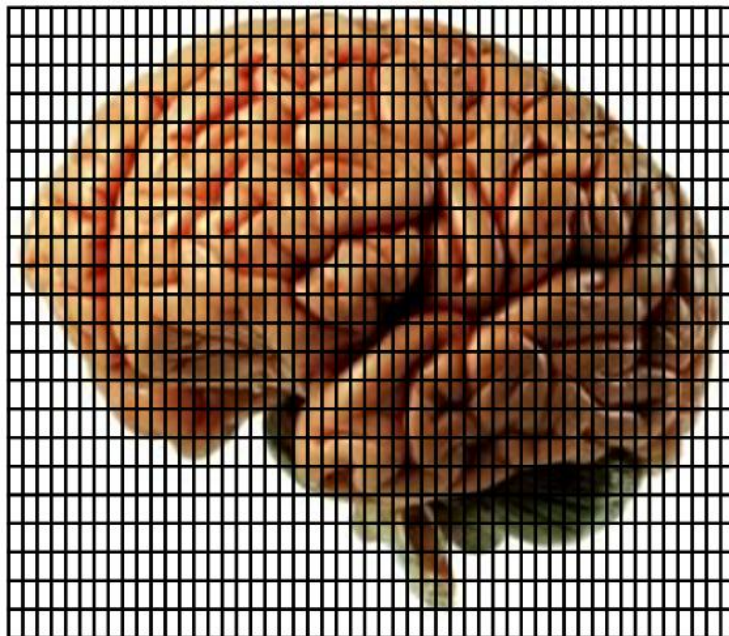
Obrázok 48 - 2D model s chybou merania väčšou ako prah

Po kliknutí na ľubovoľnú bunku rezu sa používateľovi zobrazí poloha bunky v mriežke (modrá šípka na obr. 49), hodnota markera v danom mieste (oranžová šípka na obr. 49), zaradenie do intervalu (zelená šípka na obr. 49) a hodnota chyby (šedá šípka na obr. 49). V prípade, že hodnota markera je vyššia ako stanovený limit (kritická hodnota), zobrazí sa výkričník (obr. 49) symbolizujúci miesto s extrémne zvýšenou hodnotou markera.



Obrázok 49 - Hodnoty markera

Mriežka (obr. 50) je definovaná rozlíšením magnetickej rezonancie, mozog je rozdelený na veľké množstvo buniek (veľkosť jednej bunky v 3D priestore je max. 2x2x2mm), v ktorých je meraná hodnota markera.

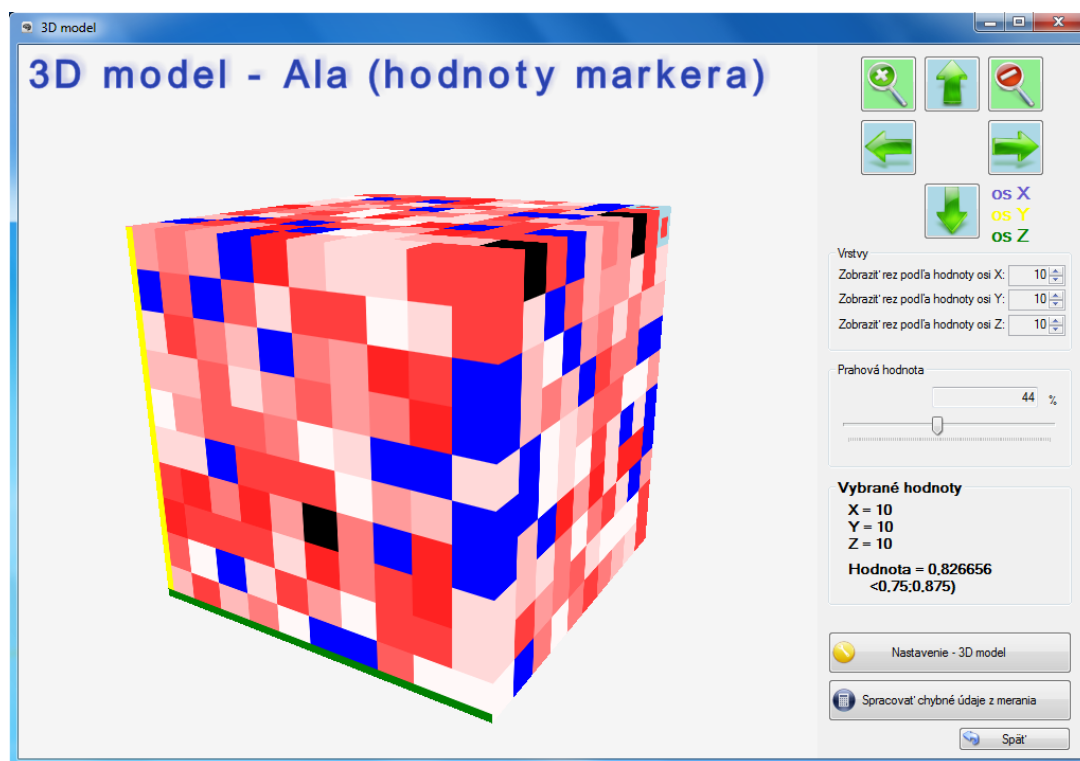


Obrázok 50 - Mriežkové meranie hodnôt

3.5.3 3D model

Iným typom prezentácie výsledkov magnetickej rezonancie sú 3D modely, ktoré spájajú všetky pohľady 2D modelov. Každú vrstvu môžeme považovať za rez kvádra. Jednotlivé rezy sú definované používateľom ľubovoľným spôsobom, s cieľom farebného zobrazenia anomálií a tumorov, ktoré predstavujú riziko pre pacienta. 3D model umožňuje zoom a rotácie akéhokoľvek rezu. Pre zobrazovanie sa využíva knižnica „XNA Game Studio 4.0 Refresh“, ktorá bola vyvinutá a je určená hlavne pre hry z dôvodu rýchlosti a stability. Knižnica umožňuje integráciu s akoukoľvek modernou technológiou.

3D model je založený na textúrach, každá hrana je textúrou, ktorej hodnoty sú vypočítavané po načítaní údajov. Model je bližšie popísaný v už spomínanej diplomovej práci Bc. J. Mešinu - 3D vizualizácia mozgu človeka. [15]

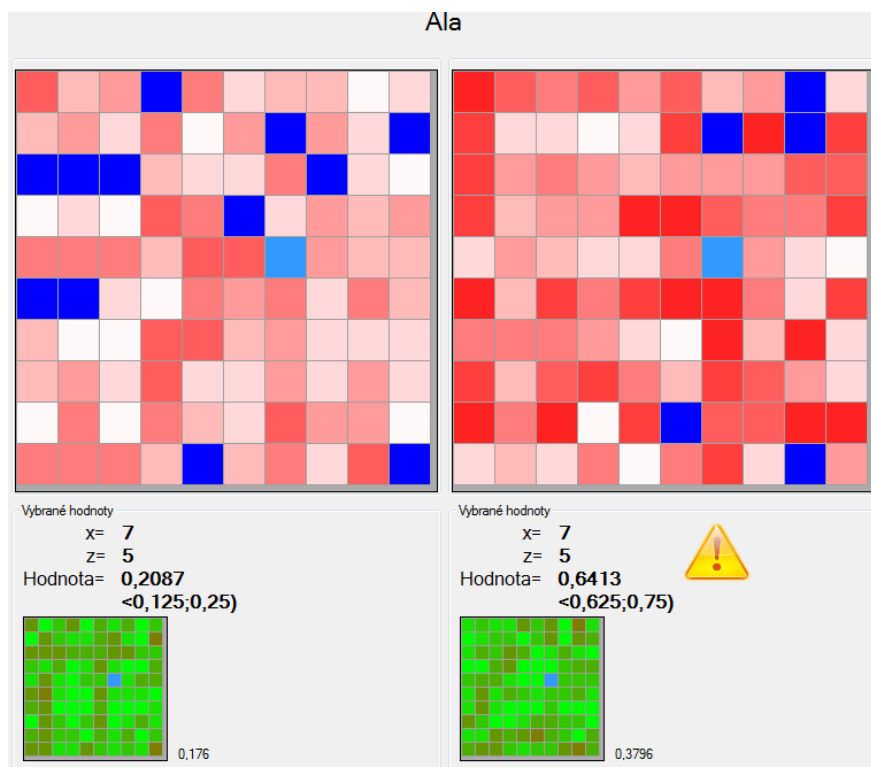


Obrázok 51 - 3D model [15]

3.6. Porovnanie výsledkov z rôznych období

V procese liečby nádorového ochorenia je dôležité sledovať progres a účinnosť liečby v čase. Ak sa nález odstraňuje operačne, je potrebné sledovať stav po operácii a ďalší vývoj. Práve z tohto dôvodu bol vytvorený modul, ktorý porovnáva výsledky vyšetrenia pacienta v rôznych časových obdobiach. Hodnota tolerancie (nastaviteľná v module *Nastavenie*) udáva maximálny rozdiel hodnôt voxelov v čase. Ak je rozdiel väčší ako tolerancia, používateľ je na túto skutočnosť upozornený. Zobrazuje sa aktuálne vybraný marker, ktorý je možné porovnať s cholínom. Princíp zobrazovania je totožný s 2D modulom. Cieľom tohto modulu, ako aj celej aplikácie, je zjednodušiť vyhodnotenie výsledkov a upriamiť pozornosť lekára na anomálie, nález a progres, resp. informovať ho, že výsledky sú bez zmeny.

Na nasledujúcom obrázku je zobrazená aktuálna situácia pacienta (napravo) a vyšetrenie pred roka (naľavo), kde ešte anomálie detegované neboli. Výkričník vyjadruje zvýšenú hodnotu markera - prekročený maximálny rozdiel hodnôt definovaný hodnotou tolerancie.



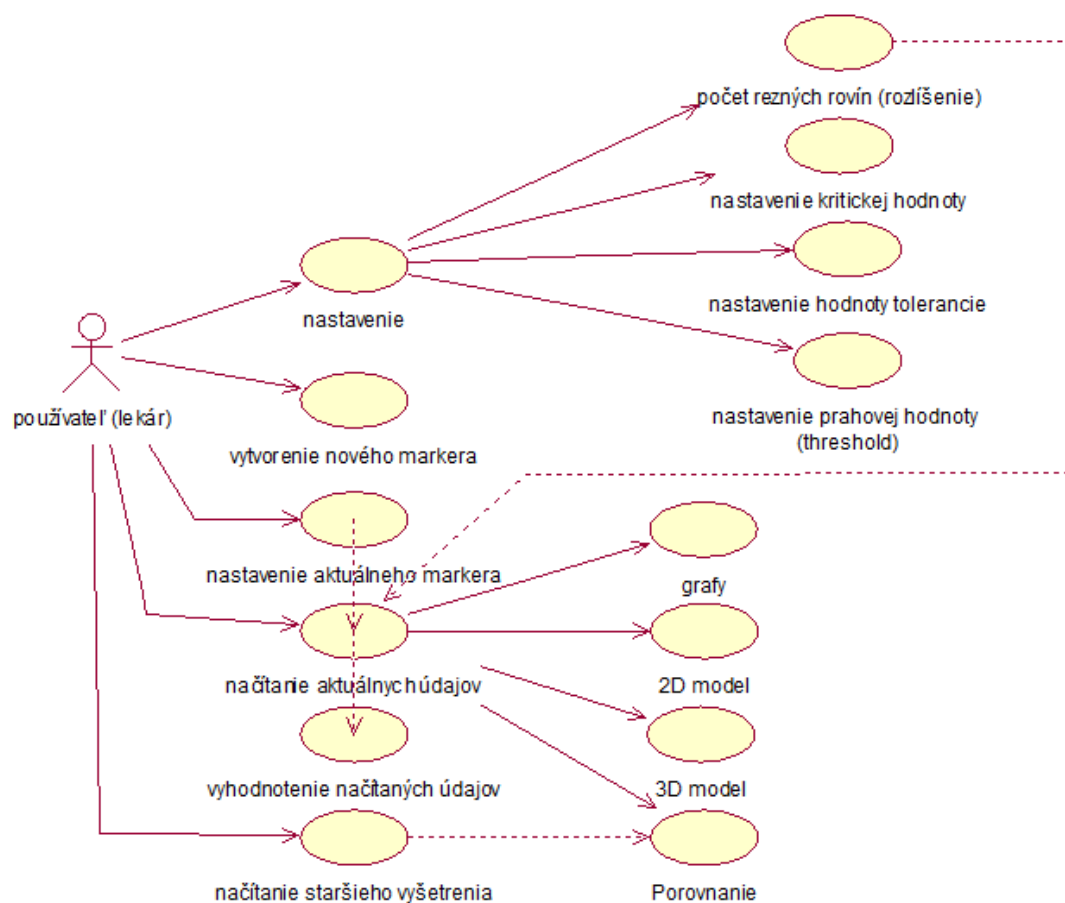
Obrázok 52 - Porovnanie výsledkov v čase

3.7 Use Case

Systém z pohľadu používateľov je rozdelený na 2 základné skupiny. Prvým aktorm je používateľ – lekár. Naš systém podporuje 26 základných markerov, z ktorých sa zobrazujú výsledky, lekár si vyberá marker, ktorý bol meraný a chce ho sledovať. Ďalšou možnosťou je nadefinovanie nového markera, ktorý vznikne aritmetickými operáciami s inými markermi.

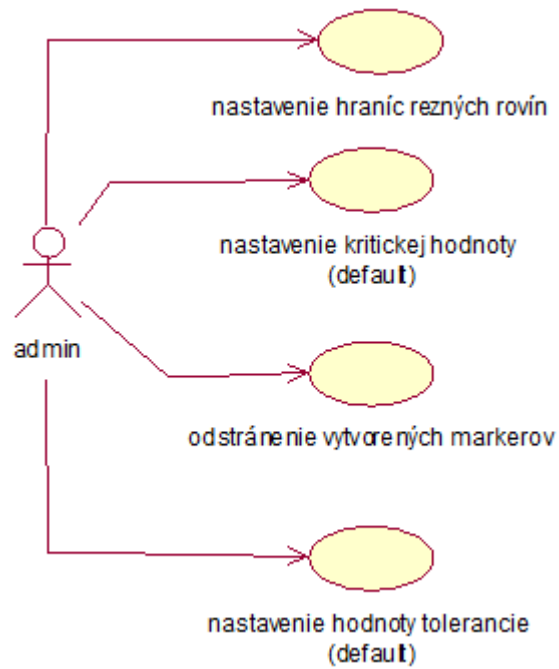
V samotnej oblasti zobrazovania výsledkov ponúkame 3 moduly – zobrazenie vo forme grafov, 2D a 3D zobrazenie.

Lekárovi je počas práce umožnená zmena počtu rezných rovín, tento počet však neovplyvňuje práve načítané výsledky. Medzi ďalšie parametre, ktoré však ovplyvňujú výsledky spracovania, je kritická hodnota, ktorá udáva maximálnu hodnotu markera, ktorú ešte nepovažujeme za nebezpečnú. Táto hodnota môže nadobudnúť akúkoľvek hodnotu z intervalu $<0 ; 1>$, nakoľko všetky načítané hodnoty sú normalizované do tohto intervalu.



Obrázok 53 - Actor Lekár (Use Case)

Druhým aktorm, ktorý v systéme vystupuje, je správca, ktorý nastavuje predvolené hodnoty počtu rezných rovín, kritickej hodnoty a hodnoty tolerancie. Jeho úlohou je aj nastaviť maximálne a minimálne počty rezných rovín na základe kvality a rozlišovacej schopnosti použitého MRI prístroja. Nastavenie aplikácie je teda priamo závislé od technických parametrov používaného prístroja. Administrátor systému môže taktiež vymazať markery, ktoré boli lekármi definované.



Obrázok 54 - Actor Admin (Use Case)

4 Výsledky

Výsledkom predkladanej záverečnej práce je medicínsky orientovaná aplikácia, ktorej cieľom je zobrazenie a spracovanie výsledkov získaných vyšetrením magnetickou rezonanciou na základe zobrazenia prítomnosti určitých špecifických markerov vyskytujúcich sa v abnormálnych nálezoch (predovšetkým tumoroch) v mozgu. Aplikácia bola vytvorená na základe konkrétnych požiadaviek praxe a je orientovaná na používateľa (lekár), ktorému umožňuje zobrazovať a spracovávať výsledky na základe jeho požiadaviek a potrieb. Aplikácia ponúka rôzne druhy zobrazenia od 2D zobrazenia cez 3D modely až po zobrazenie výsledkov vo forme grafov a datagridov. Prílohou aplikácie je používateľská príručka, obsahujúca jednoduchý návod na použitie aplikácie, ktorého hlavnou súčasťou sú názorné obrázky a schémy umožňujúce používateľovi bez technického či programátorského vzdelania aplikáciu nielen používať, ale zároveň nastaviť sledované parametre a spôsob zobrazenia podľa svojich profesionálnych požiadaviek resp. podľa požiadaviek konkrétneho pracoviska.

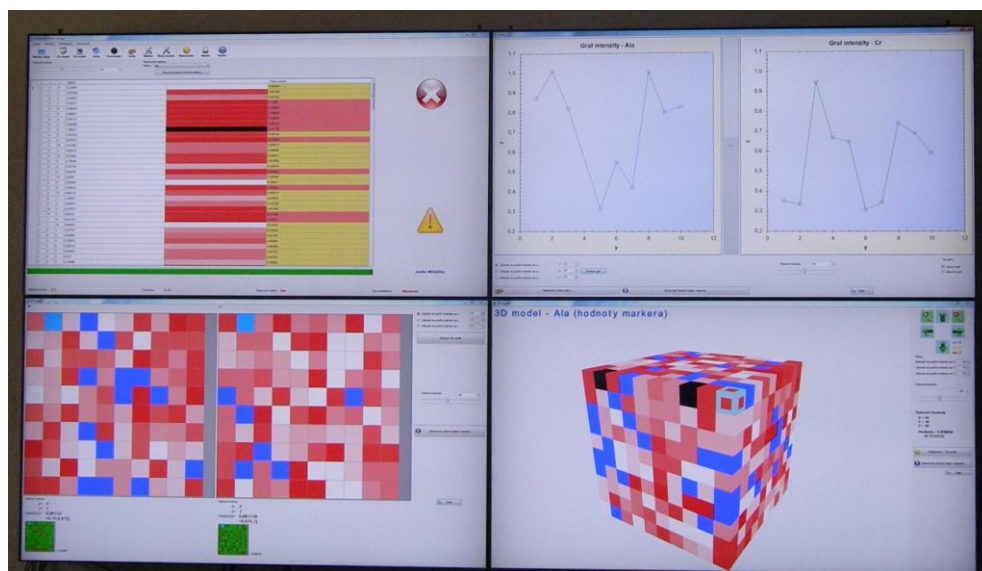
Aplikácia bola počas vývoja testovaná na výsledkoch vyšetrení dobrovoľníkov v úzkej spolupráci s Univerzitnou nemocnicou v Martine. Dobrovoľníci boli vyšetrení na jej pôde v súlade s etickými a zákonnými normami. Vyšetrenia boli realizované v štandardných podmienkach MRI prístrojom Siemens Symphony 1,5 Tesla (obr.55).



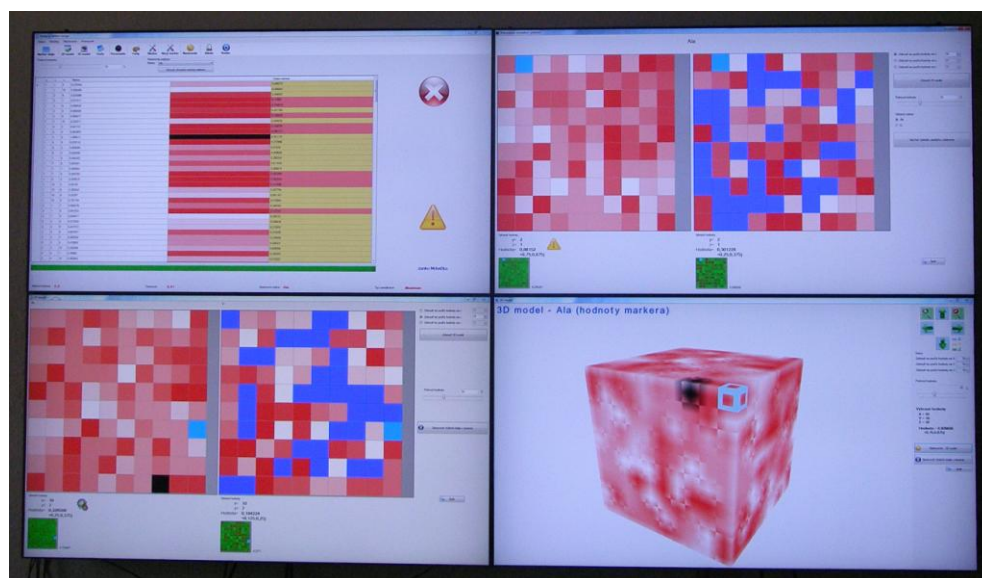
Obrázok 55 - Siemens Symphony 1,5 Tesla

Konzultácie a testovanie aplikácie boli realizované v spolupráci s Mgr. Michalom Bittsanským, PhD. na pôde Ústavu lekárskej biochémie Lekárskej fakulty v Martine, kde je zároveň aplikácia v súčasnosti v testovacej prevádzke.

Všetky pripomienky získané počas doterajšej testovacej prevádzky boli priebežne zapracovávané do aplikácie. Na základe požiadaviek bola aplikácia upravená tak, že v súčasnosti je možné jednotlivé modely zobrazovať na počítači so štyrmi monitormi, pričom údaje medzi jednotlivými oknami sa synchronizujú. Obr. 56 zobrazuje základné okno (vľavo hore) a jednotlivé modely. Obr. 57 zobrazuje 2D, 3D model a model pre porovnanie výsledkov jednotlivých vyšetrení u pacienta.



Obrázok 56 - Možnosť zobrazenia na 4 monitoroch



Obrázok 57 - Možnosť zobrazenia na 4 monitoroch

5 Záver

Onkologické ochorenia sú jedným z najzávažnejších celosvetových medicínskych problémov. Včasná diagnostika a adekvátne liečba zohrávajú kľúčovú úlohu z hľadiska ďalšej prognózy pacienta. Úlohou vedeckého a technického vývoja je prispieť svojím dielom k tomuto procesu, ktorého konečným cieľom je pacient, trpiaci závažným ochorením a snaha o záchranu jeho života, zdravia, resp. predĺženie a zlepšenie kvality jeho života.

Detekcia nádoru mozgu patrí k novým častiam projektu Nemocničný informačný systém, ktorý vznikol pre potreby Univerzitnej nemocnice v Martine na základe dlhodobej spolupráce s Fakultou riadenia a informatiky Žilinskej univerzity v Žiline.

Úlohou predkladanej diplomovej práce bolo vytvoriť univerzálne riešenie pre spracovanie výsledkov magnetickej rezonancie s cieľom detegovať a lokalizovať nádory mozgu; spracované výsledky následne vyhodnotiť a poskytnúť výsledky vo forme tabuliek, grafov, 2D a 3D modelov na základe požiadaviek používateľa. Základom aplikácie je zobraziť prítomnosť jednotlivých markerov, teda látok, ktoré sa vyskytujú v nádorovo zmenenom tkanive. Aplikácia bola koncipovaná tak, že zobrazuje jednak prítomnosť štandardných markerov, ale používateľovi umožňuje zadávať aj ďalšie markery na základe jeho odborných požiadaviek (napr. markery špecifické pre predpokladaný druh nádoru). Zároveň je možné výsledky magnetickej rezonancie ukladať a porovnávať medzi sebou v čase a sledovať tak zmenu lokálneho nálezu počas liečby, po chirurgickom výkone a pod.

Spracovanie výsledkov magnetickej rezonancie však nie je zaťažené rovnakou chybou merania vo všetkých vyšetrovaných oblastiach, niektoré časti sú merané s relatívne vysokou chybou merania. Preto bol vytvorený modul na spracovanie týchto údajov s cieľom redukovať chybu merania a získať tak presnejšiu hodnotu sledovaného markera.

V ďalšom pokračovaní projektu by bolo možné rozšíriť ho o databázu výsledkov vyšetrení spracovaných použitím predkladanej aplikácie, čím by sa získala možnosť porovnávať a kategorizovať jednotlivé typy anomálií.

Aplikácia bola vyvíjaná v spolupráci s Univerzitnou nemocnicou v Martine, jej použitie však nie je limitované konkrétnym typom prístroja a preto je možné použiť ju aj v iných nemocniciach. Verzia aplikácie aj v anglickom a nemeckom jazyku by umožňovala aj jej využitie v zahraničí.

V súčasnosti je predkladané riešenie v testovacej prevádzke na Ústave lekárskej biochémie Lekárskej fakulty v Martine.

Zoznam použitej literatúry a iných zdrojov

Knižné zdroje

- [1] Beszédes, M: Detekcia tváří a tvárových črt v obraze (dizertačná práce), 2007, Bratislava
- [2] Dobeš, M.: Zpracování obrazu a algoritmy v C#. BEN – technická literatura, 2008. ISBN: 978-80-7300-233-6.
- [3] Fabián, P.: Multimediálne informačné systémy. skriptá FRI ŽU, 1998.
- [4] Hargaš, L.: Spracovanie statickej a dynamickej obrazovej informácie pre biomedicínske aplikácie (habilitačná práca), 2010, Žilina
- [5] Hornak, J.: The Basics of MRI, Interactive Learning Software, 2008.
- [6] Horvat, L.: Digital Imaging: Essential Skills. Focal Press, 2003. ISBN: 0-240-51971-X
- [7] Jähne, B.: Digital Image Processing, Springer, 2002. ISBN: 3-540-67754-2
- [8] Kán, P.: Automatic image-based 3D head modeling (diplomová práca), 2010, Bratislava
- [9] Marček, D.: Neurónové siete a ich aplikácie. EDIS, 2006. ISBN: 80-8070-497-X.
- [10] Martincová, P. a kol.: Operačné systémy. Žilinská univerzita, 2004. ISBN: 808070242X.
- [11] Matiaško, K. a kol.: Databázové systémy a technológie. STU, 2009. ISBN: 9788022730358.
- [12] Matiaško, K. a kol.: Databázové systémy - Databázové technológie a aplikácie. EDIS, 2008. ISBN: 978-80-8070-821-4.
- [13] Matiaško, K. a kol.: Databázové systémy - Základy databázových systémov. EDIS, 2008. ISBN: 978-80-8070-820-7.
- [14] Matiaško, K. a kol.: Základy informatiky. EDIS, 2004. ISBN: 80-8070-186-5.
- [15] Mešina, J.: 3D vizualizácia mozgu človeka (diplomová práca), 2012, Žilina

- [16] Miček, J.: Šum v systénoch spracovania signálov. EDIS, 2010. ISBN: 9788054401713.
- [17] Nekula, J.: Radiologie, Univerzita Palackého, 2005. ISBN: 8024410117
- [18] Palovič, A.: Nemocničný informačný systém - Administrácia a modularita systému (diplomová práca), 2011, Žilina
- [19] Pianykh, O. : Digital Imaging and Communications in Medicine, Springer, 2008. ISBN: 978-3-540-74570-9
- [20] Piš, P.: Nemocničný informačný systém - Používateľské rozhrania (diplomová práca), 2011, Žilina
- [21] Polc, M.: Nemocničný informačný systém - Objektovo relačný dátový model pre modul ambulancia (diplomová práca), 2011, Žilina
- [22] Stevens, M.: On Collisions For MD5 (diplomová práca), 2007, Eindhoven
- [23] Teodorovič, D., Vukadinovič, K.: Traffic Control and Transport Planning: A Fuzzy Sets and Neural Networks Approach, Springer, 1998. ISBN: 079238380X
- [24] Toth, Š.: Rozpoznávanie a lokalizácia dopravných značiek (diplomová práca), 2010, Žilina
- [25] Vladár, M.: Nemocničný informačný systém - Prezentačná vrstva pre modul Ambulancia (diplomová práca), 2011, Žilina
- [26] Zeman, M.: Speciální chirurgie, Galén, 2006. ISBN: 8072622609

Internetové zdroje

- [27] <http://www.peterkan.com/diplomathesis.php> [On-line 10.3.2011]
- [28] www.computervision.com [On-line 9.1.2012]
- [29] <http://www.cs.cmu.edu/~cil/v-pubs.html> [On-line 9.1.2012]
- [30] <http://www.ai-cit.sk/source/publications/books/NS1/html/node8.html> [On-line 9.1.2012]

- [31] <http://www.ai-cit.sk/source/publications/books/NS1/html/node9.html>
[On-line 9.1.2012]
- [32] http://kplab.tuke.sk/hardwiki-mz/images/8/86/SK2010_24_prezentacia.pdf
[On-line 9.1.2012]
- [33] http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=1672890 [On-line 9.1.2012]
- [34] http://www.sapdesignguild.org/resources/glossary_color/index1.html
[On-line 25.10.2010]
- [35] <http://msdn.microsoft.com/en-us/library/bb200104.aspx>, [On-line 25.10.2010]
- [36] <http://www.npop.sk/typy-karcinomu/nadory-mozgu/> [On-line 15.12.2011]
- [37] <http://medical.nema.org/dicom/geninfo/Brochure.pdf> [On-line 15.12.2011]
- [38] <http://dicom.nema.org> [On-line 15.12.2011]
- [39] <http://data.nczisk.sk/standardy/DICOM.pdf> [On-line 15.12.2011]
- [40] http://software.intel.com/sites/products/documentation/hpc/ippi/ippi/ippi_ch6/ch6_color_models.html [On-line 12.12.2010]
- [41] http://www.aphic.com/datainflux/colour/ColorModels_adobe.pdf - On-line
[On-line 25.10.2010]
- [42] <http://frcatel.fri.uniza.sk/users/paluch/Kryptografia/HASH.pdf> [On-line 20.10.2011]
- [43] Stanek, M: Základy kryptológie -
<http://www.dcs.fmph.uniba.sk/~stanek/crypto/main2.pdf> [On-line 20.10.2011]
- [44] http://www.solen.sk/index.php?page=pdf_view&pdf_id=4455&magazine_id=10
[On-line 15.12.2011]
- [45] <http://www.mri-portal.com/verejnost.php> [On-line 15.12.2011]

Zoznam použitých skratiek

CT	Počítačová tomografia
MRI	Magnetická rezonancia
PET	Pozitrónová emisná tomografia
DICOM	Digital Imaging and Communications in Medicine
ACR	American College of Radiology
NEMA	National Electrical Manufacturers Association
ADABOOST	Adaptive Boosting
RGB	Red Green Blue (colour model)
RGBA	Red Green Blue + Alpha canal (colour model)
CMY	Cyan Magenta Yellow (colour model)
CMYk	Cyan Magenta Yellow + Black (colour model)
HSL	Hue Saturation Lightness (colour model)
HSV	Hue Saturation Value of Brightness (colour model)
MD5	Message Digest 5 (šifrovací algoritmus)

Zoznam obrázkov

Obrázok 1 - Ukážka rezu hlavy	9
Obrázok 2 - Nádor mozgu - anaplastický astrocytóm [44].....	11
Obrázok 3 - Schéma MRI [45]	14
Obrázok 4 - MRI prístroj (Siemens Avanto) [45].....	15
Obrázok 5 - Syngo fastView.....	17
Obrázok 6 - Základný komunikačný model DICOM [39]	18
Obrázok 7 - Diagram algoritmu pre nastavenie prahu [4]	20
Obrázok 8 - Lokalizácia cievneho pokrytia [4]	20
Obrázok 9 - 3D model hlavy [27].....	21
Obrázok 10 - Typy spracovania obrazovej informácie.....	22
Obrázok 11 - Neurón	23
Obrázok 12 - Matematická formulácia neurónu	24
Obrázok 13 - Neurónová sieť	24
Obrázok 14 - Algoritmus k-najbližších susedov.....	25
Obrázok 15 - RGB model [40]	27
Obrázok 16 - CMY model [6].....	28
Obrázok 17 - Princíp tlače - CMYk model [6]	28
Obrázok 18 - Odtieň [41].....	29
Obrázok 19 - Sýtosť [41]	29
Obrázok 20 - CIElab [34]	30
Obrázok 21 - Class diagram projektu detekcia_nadoru_mozgu.....	32
Obrázok 22 - Resources.....	32
Obrázok 23 - Class diagram – 3D_vizualizacia.....	33
Obrázok 24 - Spoločné triedy oboch projektov - CommonClasses.....	34
Obrázok 25 - Informácie o aktuálnom nastavení aplikácie	34
Obrázok 26 - Ukážka grafov v module Grafy	35
Obrázok 27 - Trieda Kocka	38
Obrázok 28 - Princíp merania.....	39
Obrázok 29 - Štruktúra vstupného súboru	39

Obrázok 30 - Tvorba nového markera	40
Obrázok 31 - Typ normalizácie	43
Obrázok 32 - 2D model - rez mozgu	44
Obrázok 33 - Okolie	44
Obrázok 34 - Voxely pred spracovaním	45
Obrázok 35 - Voxely po spracovaní	46
Obrázok 36 - Formulár - Spracovanie chýb.....	46
Obrázok 37 - Formulár - Nastavenie aplikácie.....	47
Obrázok 38 - Nastavenie aplikácie administrátorom systému.....	48
Obrázok 39 - Štruktúra správy.....	49
Obrázok 40 - Šifrovací algoritmus MD5 [42]	50
Obrázok 41 - Rizikové bunky neboli identifikované.....	51
Obrázok 42 - Hodnoty voxelov, ktoré prekračujú kritickú hodnotu.....	51
Obrázok 43 - Nádor a jeho okolie.....	52
Obrázok 44 - Grafy	53
Obrázok 45 - Nastavené farebné spektrum pre hodnoty markera	54
Obrázok 46 - Nastavené farebné spektrum pre chyby merania	54
Obrázok 47 - Načítanie hodnôt (diagram)	55
Obrázok 48 - 2D model s chybou merania väčšou ako prah	56
Obrázok 49 - Hodnoty markera	56
Obrázok 50 - Mriežkové meranie hodnôt	57
Obrázok 51 - 3D model [15].....	58
Obrázok 52 - Porovnanie výsledkov v čase.....	59
Obrázok 53 - Actor Lekár (Use Case)	60
Obrázok 54 - Actor Admin (Use Case)	61
Obrázok 55 - Siemens Symphony 1,5 Tesla.....	62
Obrázok 56 - Možnosť zobrazenia na 4 monitoroch	63
Obrázok 57 - Možnosť zobrazenia na 4 monitoroch	63

Zoznam rovníc

Rovnica 1 - Výpočet novej hodnoty v šedotónovom modeli.....	30
Rovnica 2 - Výpočet novej hodnoty voxelu	38
Rovnica 3 - Výpočet hodnôt nových markerov	41
Rovnica 4 - Výpočet chyby merania.....	41
Rovnica 5 - Výpočet novej hodnoty voxelu	45
Rovnica 6 - Všeobecná rovnica pre výpočet novej hodnoty z okolia ľubovoľnej šírky.....	45
Rovnica 7 - Funkcie pre šifrovací algoritmus.....	50