

Slovenská Technická Univerzita v Bratislave

Fakulta informatiky a informačných technológií

FIIT-5212-86031

František Šefčík

Spracovanie vizuálnych medicínskych dát metódami  
počítačového videnia a hlbokých neurónových sietí

Bakalárska práca

Vedúci práce: doc. Ing. Vanda Benešová, PhD.

Máj 2019

Slovenská Technická Univerzita v Bratislave

Fakulta informatiky a informačných technológií

FIIT-5212-86031

František Šefčík

Spracovanie vizuálnych medicínskych dát  
metódami počítačového videnia a hlbokých  
neurónových sietí

Bakalárska práca

**Štúdijný program:** Informatika  
**Štúdijný odbor:** 9.2.1 Informatika  
**Miesto vypracovania:** FIIT STU, Bratislava  
**Vedúci práce:** doc. Ing. Vanda Benešová, PhD.

Máj 2019



### **Čestné prehlásenie**

Čestne prehlasujem, že predloženú bakalársku prácu som vypracoval samostatne pod vedením doc. Ing. Vanda Benešová, PhD., a uviedol v zozname literatúry všetky použité odborné zdroje.

V Bratislave, 7.5.2019

.....  
František Šefčík



## **Pod'akovanie**

Moje pod'akovanie patrí najmä vedúcej práce, doc. Ing. Vanda Benešová, PhD., za jej ľudský prístup, ochotu, odborné vedenie, metodickú pomoc a cenné rady, ktoré mi poskytla počas jej vypracovania.

Osobitné pod'akovanie patrí mojim rodičom a mojim najbližším, za ich podporu, pomoc a vytvorenie podmienok pre vypracovanie bakalárskej práce.

František Šefčík



# Anotácia

SLOVENSKÁ TECHNICKÁ UNIVERZITA V BRATISLAVE

FAKULTA INFORMATIKY A INFORMAČNÝCH TECHNOLOGIÍ

Študijný program: Informatika

Autor: František Šefčík

Bakalárska práca: Spracovanie vizuálnych medicínskych dát metódami počítačového videnia a hlbokých neurónových sietí

Vedúci bakalárskej práce: doc. Ing. Vanda Benešová, PhD.

Máj 2019

Rakovina prostaty (RP) je najčastejšie diagnostikovaným typom rakoviny a dôvodom smrti spojenej s rakovinou u mužov. Počítačom podporované systémy diagnostiky (CAD) napomáhajú radiológom pri analýze snímok z multiparametrickej magnetickej rezonancie (mpMRI), za účelom rozlíšiť klinicky významné lézie prostaty od tých neškodných. Do popredia v tejto oblasti čoraz viac vstupujú metódy hlbokého učenia a umelých neurónových sietí, ktoré priniesli skvelé výsledky v oblasti spracovania obrazu. Väčšina súčasných CAD systémov pre diagnostiku RP je založených na samostatnej hlbokéj konvolučnej neurónovej sieti (CNN), ktorá je zvyčajne závislá od veľkého množstva anotovaných dát na tréning. Spoločným problémom týchto metód je nedostatok dostupných anotovaných dát, čo limituje ich úspešnosť. Pre naučenie reprezentácie obrazu s malým počtom anotovaných dát sme navrhli hlbokú Siamsku CNN architektúru, ktorá môže byť trénovaná len s binárnou informáciou o páre obrázkov.





# Annotation

SLOVAK UNIVERSITY OF TECHNOLOGY IN BRATISLAVA

FACULTY OF INFORMATICS AND INFORMATION TECHNOLOGIES

Degree course: Informatics

Author: František Šefčík

Bachelor thesis: Processing of visual medical data by computer vision methods and deep neural networks

Supervisor: doc. Ing. Vanda Benešová, PhD.

May 2019

Prostate cancer is the most common diagnosed cancer and cause of cancer-related death among men. Computer Aided Diagnosis (CAD) systems are used to support radiologists in multiparametric Magnetic Resonance (mpMR) image-based analysis, in order to distinguish clinically significant prostate cancer lesions from benign ones. The methods of deep learning and artificial neural networks, have great results in image processing, and are increasingly coming to the forefront. Most of related studies investigated CAD systems for PCa diagnosis based on single supervised convolution neural network (CNN), which usually relies heavily on a large scale annotated dataset for training. A common problem of these methods is lack of accessible annotated data which limit their approach. To learn image representation with less supervision, we propose a deep Siamese CNN (SCNN) architecture that can be trained with only binary image pair information.



# Obsah

---

<b>1</b>	<b>Úvod . . . . .</b>	<b>1</b>
<b>2</b>	<b>Analýza problematiky . . . . .</b>	<b>5</b>
2.1	Počítačové videnie . . . . .	5
2.1.1	Predspracovanie obrazu . . . . .	5
2.1.1.1	Registrácia obrazu . . . . .	5
2.1.1.2	Interpolácia . . . . .	6
2.1.2	Výpočet príznakov . . . . .	7
2.1.2.1	Príznaky intenzity . . . . .	7
2.1.2.2	Príznaky tvaru . . . . .	7
2.1.2.3	Príznaky textúr . . . . .	7
2.1.2.4	Štatistické príznaky . . . . .	7
2.1.3	Zobrazovanie magnetickou rezonanciou . . . . .	8
2.1.3.1	T2W . . . . .	8
2.1.3.2	DWI . . . . .	9
2.1.3.3	DCE . . . . .	9
2.2	Hlboké učenie a umelé neurónové siete . . . . .	9
2.2.1	Klasifikácia . . . . .	9
2.2.2	Umelá neurónová sieť . . . . .	10
2.2.2.1	Perceptrón ako obraz neurónu [8] . . . . .	10
2.2.2.2	Typy umelých neurónových sietí . . . . .	11
2.2.2.3	Viacvrstvový perceptrón (MLP) . . . . .	12
2.2.3	Konvolučná neurónová sieť . . . . .	12
2.2.3.1	Konvolučná vrstva . . . . .	13
2.2.3.2	Pooling vrstva . . . . .	14
2.2.3.3	Nelineárna vrstva . . . . .	14
2.2.3.4	Plne prepojené vrstvy . . . . .	14
2.2.4	Siamské neurónové siete . . . . .	15
2.2.5	Trénovanie neurónových sietí . . . . .	16
2.3	Súčasná riešenia . . . . .	18

2.3.1	Publikácia s názvom <i>Prostate Cancer Diagnosis using Deep Learning with 3D Multiparametric MRI</i> [18] . . . . .	19
2.3.2	Publikácia s názvom <i>Classification of Clinical Significance of MRI Prostate Findings Using 3D Convolutional Neural Networks</i> [20] . . . . .	21
2.3.3	Publikácia s názvom <i>Automated diagnosis of prostate cancer in multi-parametric MRI basedon multimodal convolutional neural networks.</i> [16] . . . . .	22
<b>3</b>	<b>Návrh . . . . .</b>	<b>25</b>
3.1	Koncept riešenia . . . . .	25
3.1.1	Predspracovanie . . . . .	26
3.1.2	Extrakcia regiónu záujmu (ROI) . . . . .	27
3.1.3	Augmentácia . . . . .	27
3.1.4	Model a klasifikácia . . . . .	28
3.1.5	Trénovanie modelu . . . . .	29
3.1.6	Metódy vyhodnocovania . . . . .	30
3.2	Použité dáta . . . . .	31
<b>4</b>	<b>Implementácia . . . . .</b>	<b>33</b>
4.1	Vývojové prostriedky . . . . .	33
4.2	Predspracovanie . . . . .	33
4.3	Trénovanie modelu . . . . .	36
4.4	Vyhodnocovanie . . . . .	38
<b>5</b>	<b>Vyhodnotenie . . . . .</b>	<b>41</b>
5.1	Experimenty . . . . .	41
5.2	Metriky . . . . .	42
5.3	Výsledky . . . . .	44
5.3.1	Experiment 1 . . . . .	44
5.3.2	Experiment 2 . . . . .	46
5.3.3	Experiment 3 . . . . .	47
<b>6</b>	<b>Záverečné zhodnotenie . . . . .</b>	<b>49</b>

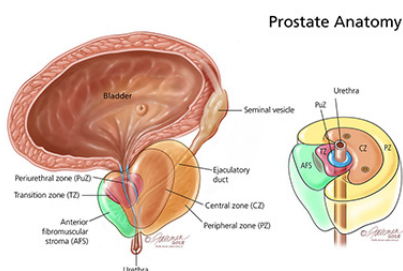
<b>Použitá literatúra</b>	<b>51</b>
<b>A Plán práce na riešení projektu</b>	<b>A-1</b>
A.1 Prvá etapa projektu	A-1
A.2 Druhá etapa projektu	A-2
<b>B Technická dokumentácia</b>	<b>B-3</b>
B.1 Používateľská príručka	B-3
B.1.1 Predspracovanie dát	B-4
B.1.2 Trénovanie modelu	B-8
B.1.3 Vyhodnotenie modelu	B-11
<b>C Článok na IIT SRC</b>	<b>C-13</b>
<b>D Opis digitálnej časti práce</b>	<b>D-19</b>



# 1 Úvod

Rakovina prostaty (RP) je druhým najčastejším druhom rakoviny a piatym najčastejším dôvodom smrti u mužov. V roku 2018 je odhadovaných 1.3 milióna nových prípadov rakoviny prostaty a 359,000 úmrtí spojených s týmto druhom rakoviny. [3] Napriek aktívnemu výskumu, zostávajú nejasnosti v dôvodoch vzniku rakoviny prostaty, avšak boli určené rizikové faktory, ktoré zvyšujú mieru jej výskytu ako genetická predispozícia, rasa, nadváha a hlavne vysoký vek. Poznáme dva druhy vývoja rakoviny prostaty, a to pomaly rastúci tumor, ktorého vývoj je pomalý a zvyčajne ostáva usadený v žľaze. Tento typ rakoviny sa vyskytuje v miere 85% zo všetkých prípadov. Druhým typom je rýchlo rastúci tumor, ktorého vývoj je rýchly a expanzívny, čiže metastázy napádajú aj iné orgány v okolí, vo väčšine sú to kosti. Metastázy v kostiach sú vysoko rizikové a vo veľkej miere ovplyvňujú úmrtnosť pacientov. Pravdepodobnosť výskytu rakoviny sa mení aj z pohľadu zonálneho rozdelenia prostaty. Až okolo 70-80% sa rakovina vyskytuje v periférnej zóne (PZ). Transitórna zóna (TZ) je postihovaná v miere 10-20%. Iba 5% sa vyskytuje v centrálnej zóne (CZ), avšak táto rakovina je agresívnejšia a prechádza aj na okolité orgány. [19]

Prostata je pohlavná žľaza mužského rozmnožovacieho systému v tvare prevrátenej pyramídy, ktorá je uložená pod mechúrom a pred konečníkom. Žľaza je zodpovedná za produkciu alkalickej tekutiny, ktorá tvorí 20-30% ejakulátu. Meria približne 3 cm na výšku a 2,5 cm na šírku, jej váha sa pohybuje od 7g do 16g u dospelých mužov. [17] Prostata je delená na tri základné zóny: CZ, ktorá tvorí 20-25% objemu, PZ je najväčšia časť so 70% a TZ, ktorá tvorí len 5% prostaty. [4]



Obr. 1: *Anatómia prostaty*<sup>1</sup>



Súčasná klinická diagnostika rakoviny prostaty sa spolieha na krvný test prostatický špecifický antigén (PSA) nasledovaný transrektálnou ultrasonografickou (TRUS) biopsiou, vďaka ktorej sa podarilo redukovať úmrtnosť o 20-30%. Avšak diagnostika pomocou spojenia PSA krvných testov a TRUS biopsie má relatívne malú presnosť, čo vedie k opakovanému počtu biopsií, ovplyvňuje úspešnosť liečby a komfort pacienta. [23] Rýchly pokrok v multiparametrickom zobrazovaní magnetickou rezonanciou (mpMRI), umožnil získať anatomické a funkčné informácie o prostate, ktoré pomáhajú zvyšovať presnosť diagnostiky RP. Ako však efektívne interpretovať údaje mpMRI na dosiahnutie uspokojivej citlivosti a presnosti pre klinické použitie zostáva nevyriešené. Využitie (PI-RADS) systému, ktorý sa snaží štandardizovať neinvazívne vyšetrenie na základe pozorovania mpMRI, síce priniesol zlepšenie, ale stále vyžaduje vysokú odbornosť a čas, ktorý je potrebný na manuálne pozorovanie mpMRI sekvencií. [29]

V poslednom storočí sa odborníci snažia o neustále zlepšovanie v oblasti zobrazovacích techník, ktoré pomáhajú radiológom k zlepšeniu detekcie a diagnostiky rakoviny. Pre diagnostiku sú stále nevyhnutní ľudia, ktorí rozhodujú o výsledku aj napriek tomu, že výsledky nemusia byť presné kvôli mnohým faktorom. Výsledky sú ovplyvnené pozorovateľom vo forme únavy, rozptýlenia, ale aj zložitou a komplexnosťou dát.

Počítačové videnie prinieslo do problematiky mnoho sľubných riešení, avšak namiesto plne automatizovaného systému detekcie rakoviny sa výskum snaží zamerať na techniky analyzovania snímok, ktoré pomáhajú radiológom pri stanovení správnych záverov. V smere tohto výskumu vznikli dva koncepty: Počítačom podporovanej detekcie (CADe) a Počítačom podporovanej diagnostiky (CADx), združené pod názvom CAD systémy. CAD je systém, ktorý kombinuje prvky umelej inteligencie a počítačového videnia s radiologickým a patologickým spracovaním snímok. Tieto techniky preukázateľne dokázali zlepšenie pri problémoch rakoviny pľúc, rakoviny prsníka alebo rakoviny hrubého čreva. Štúdie dokázali, že pri využívaní CAD systému dosiahli neskúsení radiológovia rovnaké výsledky ako radiológovia seniori, zatiaľ čo rozdiel v presnosti výsledkov bez použitia CAD systému bol výrazný. [17]

Hlboké učenie a umelé neurónové siete boli v posledných rokoch široko využité

---

<sup>1</sup>Zdroj: [https://www.urologyhealth.org/urologic-conditions/prostatitis-\(infection-of-the-prostate\).html](https://www.urologyhealth.org/urologic-conditions/prostatitis-(infection-of-the-prostate).html),

pri riešení mnohých úloh počítačového videnia, kedy je dostupné veľké množstvo anotovaných dát. [24] Pri diagnostike rakoviny prostaty je účinnosť limitovaná nedostatkom anotovaných dát a odlišnosťou od prirodzených obrázkov. Tento nedostatok sa súčasné metódy založené na hlbokom učení snažia kompenzovať rozšírením dát pomocou rôznych augmentačných techník. Augmentácia limitovaných tréningových dát pre dosiahnutie dostatočnej variancie dát je kritický krok k natrénovaniu modelu. Nesprávne prevedená augmentácia môže viesť k nesprávnym výsledkom. Jednou z ciest, ako sa vyhnúť augmentácií by mohlo byť využitie modelu neurónovej siete, ktorá nepotrebuje veľké množstvo dát na jej natrénovanie.

V tejto práci predstavíme riešenie založené na architektúre Siamských neurónových sietí (SNN). Riešenie sa zameriava na výzvu PROSTATEx, ktorá je pripravená pre úlohu klasifikácie klinicky významných nálezov rakoviny prostaty v mpMRI snímkoch. Naša SNN sa učí rozoznávať pár obrázkov s nálezmi, na základe vyhodnocovania podobnosti a vzťahov medzi nimi. Každý z páru obrázkov vstupuje do jednej z dvoch identických konvolučných neurónových sietí (CNN) a Contrastive loss funkcia je počítaná medzi výstupmi týchto dvoch sietí. V našej práci porovnáme navrhnutú SNN metódu s modelom samostatnej CNN, ktorá využíva masívne augmentačné techniky pre natrénovanie. Zameriame sa aj na analýzu jednotlivých MRI modalít, použitých pri klasifikácii.



## 2 Analýza problematiky

---

### 2.1 Počítačové videnie

Počítačové videnie je multidisciplinárna oblasť, ktorá sa zaoberá tým, ako počítače dokážu nadobúdať vysokú úroveň porozumenie obrazov alebo videa. Počítačové videnie sa využíva v rôznych úlohách spracovania, analýzy alebo chápania digitálnych obrazov. Taktiež našlo efektívne uplatnenie aj v oblasti analýzy a spracovania medicínskych dát. V tomto smere bolo vytvorených mnoho počítačom podporovaných systémov diagnostiky a detekcie anomálií v medicínskych dátach. Systémy počítačového videnia aplikované pri klasifikácii klinicky významných lézií prostaty sa skladajú z viacerých operácií vykonávaných nasledovne: predspracovanie obrazu, výpočet príznakov a klasifikácia.

#### 2.1.1 Predspracovanie obrazu

**2.1.1.1 Registrácia obrazu** je proces transformácie súboru obrazov do jednotného súradnicového systému a dá sa považovať za fúziu obrázkov, aby sa porovnateľné vlastnosti mohli ľahko spájať. Zahŕňa mapovanie bodov z jedného obrázka do zodpovedajúcich bodov iného obrázka. Pri registrácii obrázkov, geometrická transformácia mapuje umiestnenie obrázka na nové miesta, na základe referenčného obrázku.[2] Správne parametre geometrickej transformácie sú kľúčom k procesu registrácie obrázkov.

Registrácia obrázkov môže byť rozdelená do niekoľkých kategórií na základe transformačného modelu, ktorý používajú: rigidné, afinné a nerigidné transformácie. Zložitosť transformácií v každej transformácii je charakterizovaná stupňami slobody. Objekt môže byť transformovaný pomocou posunutia, rotácie alebo škálovania. [17]

Najjednoduchším typom transformácie v zmysle stupňa slobody sa zvyčajne rozumie rigidná transformácia. Tento typ transformácie sa skladá z otočenia a posunutia. Pre 2D prípad, kde  $x = (x, y) \in \mathbb{R}^2$  je rigidná transformácia  $T_R$  formulovaná ako:

$$T_R = \begin{bmatrix} \cos\theta & -\sin\theta & t_x \\ \sin\theta & \cos\theta & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 0 \end{bmatrix}$$

kde  $\theta$  je uhol otočenia a  $t_x, t_y$  reprezentujú posunutie pozdĺž osí  $x, y$ . V prípade 3D registrácie, je pridaná ďalšia zložka  $z$  ako  $x = (x, y, z)$ . Matica otočenia bude mať veľkosť  $3 \times 3$ , zatiaľ čo vektor posunutia bude obsahovať 3 hodnoty. Výsledná matica rigidnej transformácie  $T_R$  bude mať v tomto prípade veľkosť  $4 \times 4$ .

Afinná transformácia prináša ďalší stupeň slobody pridaním škálovania a rezania k otáčaniu a posúvaniu v rigidnej transformácii. Taktiež pre 2D priestor kde  $x = (x, y) \in \mathbb{R}^2$ , afinná transformácia  $T_A$  je formulovaná ako:

$$T_A = \begin{bmatrix} a_{11} & a_{12} & t_x \\ a_{21} & a_{22} & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 0 \end{bmatrix}$$

kde sú 4 parametre afinnej matice  $a_{11}, a_{12}, a_{21}, a_{22}$  a  $t_1, t_2$  posunutie. Tak ako aj pri rigidnej transformácii, aj tu bude mať matica  $T_A$  veľkosť  $4 \times 4$ , ale teraz bude mať 12 parametrov.

Poslednou skupinou transformácií je nerigidná transformácia. Vyššie prezentované modely rigidnej transformácie sú charakteristické len obmedzeným počtom jednoduchých transformácií. Avšak tieto transformácie nie sú schopné zvládnuť lokálne deformácie. V komplexných prípadoch sú potrebné nerigidné deformácie, ktoré dokážu meniť veľkosť a uhly objektu.[2]

Zvyčajne sa pri registrácii multiparametrických MRI z rovnakého protokolu využívajú len rigidné alebo afinné transformácie. Nerigidné transformácie sa využívajú skôr pri registrácii multi-protokolových obrázkoch ako histopatológie a MRI. [17]

**2.1.1.2 Interpolácia** je metóda, ktorá sa nevyhnutne spája s pojmom registrácie. V procese registrácie sa niektoré body, pixely transformujú do bodov, ktoré nie sú definované, a preto musia byť aproximované metódami interpolácie. Všeobecne sa využíva mnoho metód ako bi-lineárna interpolácia, bi-cubic a spline. Tieto metódy využívajú lineárnu interpoláciu, aby určili intenzitu pixelu, ktorý sa nachádza medzi dvoma bodmi a jeho výsledná intenzita je pomerom intenzity podľa vzdialeností od týchto bodov.

### 2.1.2 Výpočet príznakov

Výpočet príznakov je nevyhnutnou časťou k úspešnej klasifikácii objektu v obraze. Príznakom sa rozumie výsledok merania, ktorý sme dostali kvantifikovaním nejakej vlastnosti pozorovaného objektu. Viacero jednotlivých príznakov sa spája do vektoru príznakov. Práca s príznakmi uľahčuje úlohy klasifikácie a redukuje objem dát. [28]

Prízny, ktoré sú využité vo väčšine CAD systémoch diagnostiky RP sú založené na intenzite, textúre, štatistike a tvaroch. [25]

**2.1.2.1 Prízny intenzity** sú najčastejším používaným typom príznakov, pretože RP ovplyvňuje intenzitu signálu na mp-MRI snímkoch. Princíp výpočtu spočíva v extrakcii intenzity z MRI modality, ktorá nás zaujíma.[17]

**2.1.2.2 Prízny tvaru** taktiež využívajú informácie o zmene intenzity signálu, ale poskytujú doplňujúcu informáciu o prechode intenzity signálu. Každý príznak je vypočítaný prechodom filtra hrán cez pôvodný obrázok. Výsledkom je redukcia pôvodného obrázka len na hrany, ktoré môžu detekovať tvar objektu alebo jeho štruktúru.[17]

Gáborové filtre sú často alternatívou k bežným detektorom hrán, ich výhodou je aj možnosť meniť smer a hustotu filtrov pre získanie špeciálnych príznakov. Vytvorením banky gáborových filtrov s rozdielnymi parametrami môžeme charakterizovať aj textúry.

**2.1.2.3 Prízny textúr** poskytujú ďalšiu charakteristickú vlastnosť pre identifikáciu objektu alebo oblasti záujmu v obraze. Textúra je rozpoznateľná variancia intenzít v obraze, ktorá sa dá opísať a dajú sa podľa nej určiť hranice regiónu. Zmena intenzity môže mať prevládajúci smer, môže sa opakovať alebo byť nepravidelná.[28]

**2.1.2.4 Štatistické príznaky** sú ďalším typom charakteristík obrázka, ktoré sa už nezameriavajú osobitne na jednotlivé pixely, ale na celý pozorovaný región. Najčastejším typom sú štatistické momenty ako priemer, štandardná odchýlka, exces a nesúmernosť. Niektoré štúdie extrahujú body založené na percentyloch.[17]

### 2.1.3 Zobrazovanie magnetickou rezonanciou

Z dôvodu hlbšieho objasnenia problematiky vo využití multiparametrických MRI venujeme osobitne jednu časť zobrazovacím technikám magnetickej rezonancie.

Zobrazovanie magnetickou rezonanciou (MRI) je jednou zo zobrazovacích modalít, ktorá využíva neionizujúce žiarenie na získanie užitočných diagnostických snímok. Pacient je pri tomto druhu zobrazovania vložený do veľmi silného homogénneho magnetického poľa. Do jeho tela je vyslaný krátky rádiový impulz a po jeho skončení sa sníma slabý signál, ktorý vytvára pacientovo telo, a ktorý sa následne použije na rekonštrukciu samotného obrazu.

MRI má veľký potenciál na zlepšenie procesu diagnostiky rakoviny prostaty. Až do nedávnej doby bola presnosť MRI pri lokalizácii a detekcii rakoviny prostaty obmedzená, a preto nebola ani bežne zavedená do klinickej starostlivosti. MRI začalo byť významne populárne až s využitím multiparametrického prístupu, čo je kombinovanie viacerých techník MR zobrazovania. V tomto prístupe sa jedná o kombinovanie T2-vážených MRI (T2W) s najmenej dvoma ďalšími, ako difúzne vážené zobrazovanie (angl. diffusion weighted imaging DWI) a dynamické zobrazovanie so zvýšeným kontrastom (angl. dynamic contrast-enhanced DCE). [6]

Jednou z výhod MRI je, že snímky môžu byť zaznamenané v troch rovinách bez toho, aby sa musel pacient pohnúť. Tieto roviny su prezentované ako:

**Sagittal** rovina alebo stredná rovina, ktorá rozdelí telo na pravú a ľavú časť.

**Coronal** rozdelí telo na prednú a zadnú časť, inak povedané vznikne rovina x-y osi.

**Transverse** tiež známa ako horizontálna rovina, je rovnobežná so zemou a rozdeľuje telo na hornú a dolnú časť.

**2.1.3.1 T2W** zobrazovanie ponúka výborný kontrast mäkkých tkanív a výborný pohľad na zónové zobrazenie anatómie prostaty. T2W zobrazovanie je najlepšie na identifikáciu defektov v anatómii prostaty spôsobené napríklad bunkami rakoviny. [22] Zdravé tkanivo prostaty je extrémne bohaté na vodu s riedko prepletenou hladkou svalovinou, čo poskytuje jasný, vysoko intenzívny vzhľad. Na druhej strane tkanivo rakoviny sa javí ako zlé zaostrenie zo slabého signálu, ktoré je v kontraste s vysokou intenzitou normálneho tkaniva prostaty.

**2.1.3.2 DWI** RP je charakterizovaná vysokou bunkovou hustotou, ktorá obmedzuje náhodný Brownov pohyb molekúl vody v extracelulárnom priestore. DWI je funkčná technika, ktorá meria Brownov pohyb molekúl. Rakovina sa na snímkoch javí ako jasne hyperintenzívne oblasti, pretože obmedzená difúzia vody v týchto oblastiach spôsobuje stratu signálu. DWI sa využíva na vytvorenie mapy zdanlivo difúzanho koeficientu (ADC), kde sa rakovina javí ako tmavé škvrny a taktiež dobre predpovedajú agresivitu nádoru. [22]

**2.1.3.3 DCE** sú vytvárané sériovým získavaním T1W snímkov pred, počas a po intravenózne iniekcii kontrastných látok. DCE snímky pomáhajú zvyšovať špecifickosť predikcie v porovnaní so samotnými T2W. DCE dokážu odhaliť nádor na základe zvýšeného prietoku krvi, ktorý je spôsobený zväčšením siete krvných ciev, ktoré vyživujú nádor. [22]

## **2.2 Hlboké učenie a umelé neurónové siete**

Veľký prínos do oblasti počítačového videnia a spracovania obrazu prinieslo v poslednom desaťročí hlboké strojové učenie a neurónové siete. Hlboké učenie umožňuje výpočtovým modelom naučiť sa reprezentácie údajov s viacerými úrovňami abstrakcie, a tak riešiť aj problémy, na ktoré už bežné metódy pri hlbkej štruktúrovanosti dát nepostačujú. Aby sme boli schopní pochopiť komplexné hlboké neurónové siete využívané pri problémoch počítačového videnia, budeme musieť najskôr zadať, čo je to neurónová sieť, z čoho sa skladá a aké sú jej úlohy.

### **2.2.1 Klasifikácia**

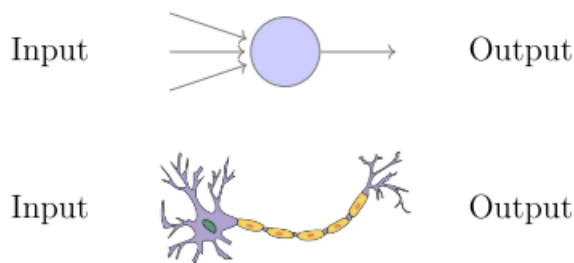
Predtým, ako sa pustíme do vysvetľovania umelých neurónových sietí je vhodný krátky úvod do problematiky klasifikácie. Klasifikácia je problém, kedy sa snažíme pozorovaný objekt na základe vstupných atribútov zaradiť do triedy, ktorej patrí, čiže priradiť mu štítok. Napríklad na základe pozorovaných medicínskych snímkov klasifikovať, či je tumor na obrázku zhubný alebo nezhubný.

Ak sa umelé neurónové siete využívajú pri probléme klasifikácie, tak sú výstupy klasifikácie reprezentované ako pravdepodobnosti vstupov patriacich do každej kategórie.

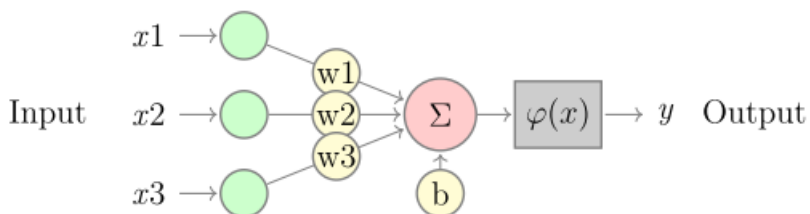


### 2.2.2 Umelá neurónová sieť

Umelá neurónová sieť (angl. ANN - Artificial neural network) je inšpirovaná ľudským mozgom, ako to vyplýva aj z časti jej názvu neurónová. Medzi ľudskou neurónovou bunkou (neurónom) a uzlom v neurónovej sieti je mnoho podobností. Neurón ľudského tela prijíma pomocou dendritov ako vstup signály, ktoré ak presiahnú určitú úroveň, tak neurón vygeneruje impulz, ktorý sa šíri na ďalšie neuróny prostredníctvom axónu. [8] ANN neurón pozostáva z vážených uzlov ako vstup a aktivačnou funkciou ako výstup. Podobnosť medzi oboma neurónmi môžete vidieť na obrázku 2.



Obr. 2: Ukážka podobnosti medzi neurovou bunkou a neurónom umelej neurónovej siete.



Obr. 3: Schématická reprezentácia jednoduchého perceptrónu.

**2.2.2.1 Perceptrón ako obraz neurónu [8]** je najmenší možný model neurónovej siete, ktorá pozostáva len z jedného neurónu. Ako môžeme vidieť na obrázku 3, perceptrón prijíma vstupné signály  $\bar{x} = (x_1, x_2, \dots, x_{n+1})$  cez synaptické váhy tvoriace váhový vektor  $\bar{w} = (w_1, w_2, \dots, w_{n+1})$ . Zložky vstupného vektora môžu nadobúdať

reálne alebo binárne hodnoty. Výstup perceptrónu je daný vzťahom:

$$y = \varphi(\xi) = \varphi\left(\sum_{j=1}^n w_j x_j - b\right)$$

kde premenná  $\xi$  označuje vnútorný potenciál neurónu, počítaný ako vážený priemer vstupných hodnôt, počet neurónov je rovný  $n$ ,  $b$  je prah (angl. bias) a  $\varphi$  volá sa aktivačná funkcia perceptrónu. Najznámejšími aktivačnými funkciami pre bežné ANN sú sigmoida alebo lineárna funkcia. V praxi sa využíva ReLu funkcia, ktorá je definovaná:

$$\varphi(\xi) = \begin{cases} 0 & : \xi \leq 0, \\ 1 & : \xi > 0. \end{cases}$$

#### 2.2.2.2 Typy umelých neurónových sietí

**Dopredné neurónové siete** (FFNN - Feed forward neural networks) sú veľmi priamočiaré, informácie sú spracúvané odpredu smerom na koniec (vstup, výstup). Neurónové siete sú zložené z vrstiev, ktoré majú svoje vstupy a výstupy. FFNN nepoznajú spätné väzby, v ktorých sú výstupy danej vrstvy vrátené do seba. Vo všeobecnosti sú všetky susedné vrstvy plne prepojené (neurón jednej vrstvy je prepojený so všetkými neurónmi susednej vrstvy).[27] V základe sa jedná o extrahovanie informácií zo vstupu pre klasifikačné alebo regresné úlohy.

**Rekurentné neurónové siete** (RNN - Recurrent neural network) sú NN, ktorých neuróny majú prepojenia sami so sebou. Neurón neprijíma len informáciu z predchádzajúcej vrstvy, ale aj sám od seba z predchádzajúceho prechodu. To znamená, že sieť nie je ovplyvňovaná len tým, čo práve spracováva, ale aj tým, čo spracovala predtým. Tento typ NN sa využíva hlavne v úlohách, kde je podstatný kontext, keď rozhodnutie v predchádzajúcich iteráciách alebo prípadoch ovplyvňuje súčasné. Najčastejším prípadom kontextu sú texty, kedy môže byť slovo analyzované len v kontexte predošlých slov alebo významu.[27]

**Autoenkóder** je podobný FFNN, odlišuje sa od nich skôr vo využití ako v architektúre. Základnou myšlienkou je natréňovanie siete automaticky, aby

výstup siete bol kópiou vstupu. Skryté vrstvy siete sa snažia naučiť reprezentáciu vstupu, snažia sa o kompresiu. Najmenšie vrstvy sú vždy v strede a informácie v nich sú najviac komprimované. Celá sieť je symetrická podľa strednej vrstvy.[27] Trénovanie siete prebieha pomocou spätného šírenia, kde chyba je vypočítaná ako rozdiel medzi vstupom a výstupom. Pri využití tejto siete sa zoberie výstup zo strednej vrstvy, ktorý je aproximáciou vstupu.

**Konvolučné neurónové siete** sa trochu odlišujú od iných sietí. Primárne sú zamerané na spracovanie obrazu, ale môžu byť využité aj pri spracovaní zvuku. [27] Zjednodušene tieto siete spracujú obrázok a klasifikujú ho. Konvolyčným sieťam venujeme ešte osobitný priestor v stati 2.2.3, a preto v kontexte CNN spomenieme ešte *Dekonvolučné siete*, ktoré sú obrátené CNN. Zmyslom tejto siete je naučiť sa napríklad zo zadaného slova vyprodukovať obrázok.

**2.2.2.3 Viacvrstvový perceptrón (MLP)** je jednou z tried neurónových sietí s dopredným šírením. Pospájaním viacerých neurónov môžeme vytvoriť sieť, kde výstup jedného neurónu môže byť vstupom iného. Takto dokážeme vytvárať vrstvy, ktoré sú zložené z perceptrónov a každý perceptrón jednej vrstvy je prepojený s perceptrónmi v susedných vrstvách. Stretávame sa tu s pojmami Hlboké neurónové siete alebo Hlboké učenie, ktoré vychádza z podstaty MLP, ktorý obsahuje okrem vstupnej a výstupnej vrstvy minimálne ešte jednu skrytú vrstvu. S pridávaním ďalších vrstiev, tak zväšujeme hĺbku neurónovej siete.

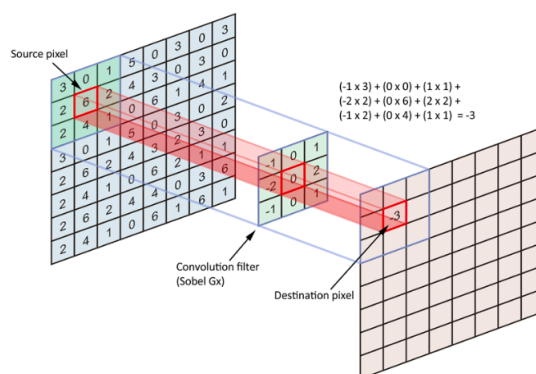
### 2.2.3 Konvolyčná neurónová sieť

Konvolučné neurónové siete (CNN z anglického convolutional neural network) sú špeciálnym typom neurónových sietí, ktoré sú široko využívané v oblasti počítačového videnia. CNN boli aplikované v mnohých úlohách klasifikácie alebo sémantickej segmentácii obrazu.

Existuje niekoľko dôvodov, prečo sú CNN dôležité. V tradičných modeloch je potrebné ručne navrhovať extraktory črt, ktoré rozpoznávajú vzory. Výhodou pri CNN je, že v procese tréningu sa určujú váhy konvolyčných vrstiev, ktoré slúžia ako extraktory črt.[11] Ďalšia výhoda je vo vylepšenej štruktúre CNNs siete, ktorá vedie k úsporám požiadavok na pamäť a požiadavok na zložitosť výpočtu.

V konvolučnej neurónovej sieti majú skryté vrstvy rôzne účely a rôzne konštelácie. Skladaním viacerých rozdielných vrstiev dokážeme vybudovať komplexnú architektúru, ktorá dokáže riešiť náš klasifikačný problém. Väčšina z týchto vrstiev je voliteľná, avšak aby sme mohli túto sieť nazývať konvolučnou, musí obsahovať aspoň jednu konvolučnú vrstvu. Pre CNN sú najbežnejšie tieto vrstvy:

**2.2.3.1 Konvolučná vrstva** Operácia konvolúcie extrahuje rôzne črty zo vstupného obrázku. Prvá konvolučná vrstva extrahuje nízko-úrovňové črty ako uhly, línie a rohy. Následne ďalšie konvolučné vrstvy extrahujú zložitejšie črty. Operácia konvolúcie je aplikovaná nad vstupnými dátami za použitia filtra alebo kernelu (jadra) k vyprodukovaniu mapy (matice) príznakov. [5]



Obr. 4: Operácia konvolúcie nad vstupnými dátami<sup>2</sup>

Ako môžeme vidieť na obrázku 4, filter sa posúva nad vstupom. Na každom mieste sa vykoná násobenie matice a súčet výsledkov sa zapíše do matice príznakov.

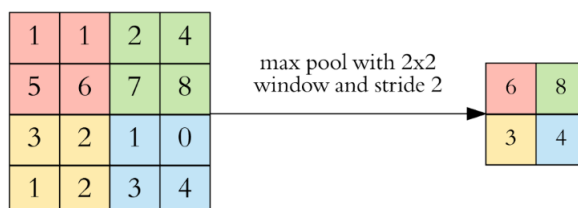
Po aplikovaní konvolúcie vzniká mapa príznakov, ktorá je vždy menšia ako vstup. Môžeme si to všimnúť aj na obrázku 4, kde vznikajú okraje bez vyplnených hodnôt. Aby sme predišli zmršťovaniu mapy príznakov, pridáva sa vrstva nulových pixelov okolo nášho vstupu.

Nad vstupmi vykonávame množstvo konvolúcií, pričom každá operácia využíva iný filter. Výsledkom sú rôzne matice príznakov. Na koniec všetky tieto matice spojíme dokopy ako výstup jednej konvolučnej vrstvy.

<sup>2</sup>Zdroj: <https://medium.freecodecamp.org/an-intuitive-guide-to-convolutional-neural-networks-260c2de0a050>,

**2.2.3.2 Pooling vrstva** sa využíva na zmenšenie dimenzionality. To nám umožňuje znížiť počet parametrov, čo skracuje tréningový čas a bojuje proti pretrénovaniu.

Najznámejším typom pooling vrstvy je max pooling, ktorý zjednodušene zoberie najväčšiu hodnotu v pooling okne. Podobne ako pri konvolúcii sa okno posúva po vstupe a dopredu musíme definovať veľkosť okna a jeho posun. [5]



Obr. 5: Max pooling získava maximálne hodnoty<sup>3</sup>

**2.2.3.3 Nelineárna vrstva** je vrstva konvolučnej siete, ktorá prepočíta vstup z konvolučnej alebo pooling vrstvy pomocou nelineárnej aktivačnej funkcie. Najznámejšou takouto funkciou v kontexte CNN je rectified linear, ktorá je definovaná ako:

$$f(x) = \max(x, 0)$$

kde  $x$  je vstup do neurónu. ReLU (rectified linear unit) je vrstva, ktorá využíva túto funkciu ako svoju aktivačnú funkciu. Výhodou ReLU oproti ostatným nelineárnym vrstvám je, že sieť sa trénuje mnohokrát rýchlejšie.[11]

**2.2.3.4 Plne prepojené vrstvy** sa zvyčajne používajú na ukončenie CNN. Tieto vrstvy však dokážu pracovať len s jednorozmernými numerickými dátami. Preto je najskôr potrebné výstup z konvolučnej alebo pooling vrstvy zmeniť z 3D do 1D. Táto časť CNN je rovnaká ako bežné ANN.

Výstup zo siete môže byť ťažko interpretovateľný. Pri klasifikačných problémoch býva zvykom ukončiť CNN softmax funkciou:

<sup>3</sup>Zdroj: <https://medium.freecodecamp.org/an-intuitive-guide-to-convolutional-neural-networks-260c2de0a050>,

$$\tilde{y}_i = \frac{e^{y_i}}{\sum_j e^{y_j}}$$

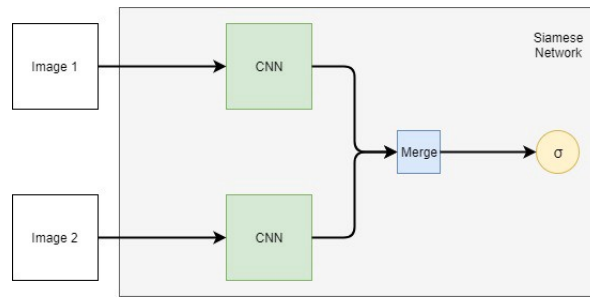
Táto normalizácia prevádza súčet vstupov na jeden, a preto môžeme interpretovať výstup softmax funkcie  $\tilde{y}_i$  ako pravdepodobnosť, či vstup patrí do triedy  $i$ .

#### 2.2.4 Siamské neurónové siete

Pre lepšie pochopenie významu siamských sietí je potrebné si ozrejmiť, čo je to One-shot learning. Vo väčšine algoritmov hlbokého učenia potrebujeme veľké množstvo dát a čím viac ich máme, tým sú naše výsledky lepšie. Avšak v niektorých prípadoch sme brzdení práve nedostatkom označovaných dát. V tomto prípade sa práve algoritmus One-shot learning zameriava na klasifikovanie objektu, na základe jedného alebo len zopár tréningových dát. [7]

Siamské neurónové siete sú špeciálnym typom neurónových sietí, ktoré sú postavené na princípe One-shot learning alebo v našom prípade konkrétnejšie one-shot image recognition. Tieto siete sa skladajú z dvoch alebo viacerých identických sietí, čo znamená, že majú rovnakú konfiguráciu s identicky nastavenými parametrami a váhami. Aktualizácia váh a parametrov prebieha vo všetkých podsieťach zrkadlovo. Tento druh architektúry sa využíva pri hľadaní podobností alebo vzťahu medzi dvoma porovnateľnými objektami. [13]

Ako sme spomenuli vyššie, Siamské siete sa skladajú z viacerých podsietí. V prípade rozpoznávania objektu z obrazu sa ako podsiete využívajú konvolučné neurónové siete, ktorých úspešnosť pri rozpoznávaní obrazu sme už uvádzali. [31, 12] Dve identické podsiete spracujú dva vstupné obrázky, ďalší modul spracuje výstup týchto podsietí do výsledného výstupu.



Obr. 6: Architektúra konvolučnej siamskej siete

Na obrázku 6 môžeme vidieť architektúru siamskej siete s dvoma konvolučnými podsieťami. Vstupom sú dva rozdielne obrázky. Prvá CNN získa z prvého obrázka vektor príznačkov, následne identická CNN získa z druhého obrázka ďalší vektor príznačkov. Aby sme porovnali podobnosť dvoch obrázkov, je vypočítaný rozdiel medzi ich vektormi príznačkov ako:

$$d(\text{Image1}, \text{Image2}) = ||f(\text{Image1}) - f(\text{Image2})||^2$$

Cieľom trénovania siamských sietí je dosiahnuť, aby rozdiel medzi vektormi príznačkov pre identické objekty bol čo najmenší, a naopak pre rozdielne objekty čo najväčší.

### 2.2.5 Trénovanie neurónových sietí

Po tom ako zostavíme našu neurónovú sieť, nastavením rôznych parametrov, vybratím vhodnej aktivačnej funkcie alebo, aké a koľko vrstiev použijeme, môžeme sa pustiť do procesu trénovania. Trénovanie neurónovej siete je proces, pri ktorom upravujeme váhy na jej vrstvách za cieľom dosiahnutia korektných výsledkov. Na tento účel sa využíva metóda nazývaná spätné šírenie chyby (angl. backpropagation).[15] Spätné šírenie chyby je metóda na výpočet gradientu chybovej funkcie (angl. loss function), pričom zohľadňuje parametre neurónovej siete. [21]

Chybová funkcia (účelová funkcia) počíta rozdiel medzi výstupom  $x_o$  a očakávaným výstupom  $\hat{x}_o$ , po tom ako bol trénovaný objekt prehnaný sieťou.

$$E = \frac{1}{2}(x_o - \hat{x}_o)^2$$

Zjednodušené nám určuje, ako veľmi sa naša sieť mylí, a preto sa snažíme o minimalizovanie tejto chyby. Na to, aby sme však upravili túto chybu, musíme zmeniť parametre siete, na čo sa najčastejšie využívajú tzv. gradientové metódy, založené na použití gradientu chybovej funkcie pre iteratívnu konštrukciu optimálneho riešenia. [8]

Gradient descent je optimalizačný algoritmus, ktorý sa snaží nájsť parametre siete na minimalizovanie chybovej funkcie. Jeho snahou je taktiež urobiť jednotkový krok v smere najväčšieho klesania ako

$$w_{jknew} = w_{jk} + \Delta w_{jk}, kde \Delta w_{jk} = -\eta \frac{\partial E}{\partial w_{jk}}$$

kde  $\eta$  je veľkosť kroku, ktorý je všeobecne využívaný pri ANN ako parameter rýchlosti učenia. [21]

Poznáme tri varianty tohto algoritmu, ktoré sa rozlišujú v tom, koľko tréovacích akcií využívajú pri počítaní gradientu. Prvým typom je Batch gradient descent, ktorý urobí výpočet gradientu pre celý tréovací dataset, preto je veľmi pomalý a je nepredstaviteľné vložiť celý dataset do pamäte. Druhým typom je Stochastic gradient descent, ktorý uskutočňuje aktualizáciu parametrov v každom tréovacom kroku. Tento prístup je rýchlejší, ale vykazuje väčšie výkyvy pri hľadaní minima. [21]

V závislosti od množstva údajov je najlepšie uskutočniť kompromis medzi presnosťou aktualizácie parametrov a časom potrebným na vykonanie aktualizácie. [21] Mini-batch gradient descent berie výhody oboch hore uvedených prístupov a uskutočňuje aktualizáciu parametrov každých  $n$  tréovacích krokov ako:

$$\Delta w_{jk} = -\eta \sum_n \frac{\partial E}{\partial w_{jk}}$$

Keď už vieme ako našu sieť natrénovať, je potrebné korektne vyhodnotiť jej úspešnosť. Vyhodnotiť úspešnosť môžeme na dátach, ktoré slúžili na natrénovanie siete. Tento prístup nemusí viesť k relevantným výsledkom z dôvodu, že sa sieť mohla naučiť tieto dáta naspamäť. Na korektné vyhodnotenie úspešnosti potrebujeme náš dataset rozdeliť na tréovaciu sadu, na ktorej budeme sieť trénovať a testovaciu, na ktorej overíme jej úspešnosť. V procese tréovania sa však často stretávame s javom, že úspešnosť na tréovacej vzorke je vysoká, zatiaľ čo na testovacej je



výrazne horšia. Tomuto javu hovoríme pretrénovanie siete, čo znamená, že naša sieť sa naučila predikovať výsledky naspamäť. Vychádza z podstaty učenia neurónovej siete, ktorá sa snaží minimalizovať chybovú funkciu vzhľadom na dáta, ktoré pozná.

Existuje niekoľko riešení, ktoré sa snažia zabrániť pretrénovaniu alebo pomáhajú vytvoriť dostatočne všeobecný model, ktorý pracuje správne aj nad dátami, ktoré neboli použité pri tréningu.

**Cross-validation** je technika, kedy rozdelíme náš dataset na tréningovú, validačnú a testovaciu množinu. Aby nedošlo k pretrénovaniu siete, využijeme validačnú množinu, aby sme vystihli moment, kedy zastavíme tréning. Najlepšia úspešnosť je v momente, keď sieť dosahuje najlepšiu úspešnosť na tréningovej a zároveň na validačnej. [15]

**Dropout** je technika, kedy počas tréningu sú deaktivované náhodné neuróny siete. Výsledkom je redukovať tendenciu siete spoliehať sa na niektoré neuróny, pretože nemusia byť stále dostupné. [15]

**Rozšírenie datasetu** je jedno z riešení, ktoré sa využíva v prípade, ak náš dataset obsahuje málo dát k natréningu dostatočne všeobecného modelu. Získať väčšie množstvo dát je často zložité, až nemožné. Takže jedným z riešení je umelo rozšíriť dataset. V praxi sa využíva pridanie šumu, mierne zväčšenie alebo zmenšenie obrázkov, otáčanie atď. Tieto upravené obrázky sú pridané do datasetu s rovnakým označovaním ako pôvodný obrázok.

## 2.3 Súčasné riešenia

Výskum v oblasti diagnostiky rakoviny prostaty priniesol mnoho sľubných riešení. Všeobecne povedané, existujú metódy formulujúce diagnózu RP ako sériu klasifikačných úloh založených na učení s učiteľom t.j. klasifikátor využíva súbor anotovaných vzoriek prostaty s ich štítkami indikujúcimi prítomnosť/neprítomnosť RP alebo agresivitu lézie, ak existuje. Existujúce metódy sa navzájom rozlišujú v získavaní príznakov a klasifikátoroch, ktoré reprezentujú alebo rozlišujú tkanivom prostaty na obrázku, v rozdielnom kombinovaní modalít alebo metódach ich fúzie.

Súčasné počítačom podporované systémy diagnostiky (CAD) analyzujú snímky rôznych zobrazovacích modalít, avšak najviac využívaným sa stalo multiparamet-

rické MRI. [18, 20, 1, 16] V tomto smere prišlo veľa zaujímavých riešení založených na hlbokých konvolučných sieťach, ktoré demonštrovali ich efektívne využitie pri úlohach počítačového videnia s dostatočným počtom anotovaných dát pre tréning. [24, 10, 26]

Množstvo dát je kritický faktor pri trénovaní hlbokých neurónových sietí. Problém s nedostatkom dát je v medicínskej oblasti obzvlášť prítomný, preto sa zainteresované organizácie snažia kvôli potrebe výskumu tieto dáta zbierať a uverňujú ich v rámci rôznych výziev. Výzva s názvom PROSTATEx<sup>4</sup> od organizácie SPIE poskytuje 3D MRI dáta rakoviny prostaty za účelom výskumu v diagnostike klinicky signifikantných nálezov. Táto výzva je zdrojom dát pre práce prezentované v tabuľke 1. Práce sa zaoberali rozlíšením medzi klinicky signifikantnou a neškodnou RP. Týmto smerom sa bude uberať aj naša práca, a preto sa v nasledujúcich podkapitolách detailnejšie pozrieme na práce z tabuľky 1.

Autor	Model	Modality	Augmentácia	AUC
Liu [18]	CNN + softmax	T2W+DWI+ADC+Ktrans	3D rotácie, rezy	0,89
Le [16]	CNN + SVM	T2W+ADC	rigid, non-rigid	0,909
Mehrt.[20]	CNN + softmax	ADC+BVAL+Ktrans	obracanie, translácie	0,85

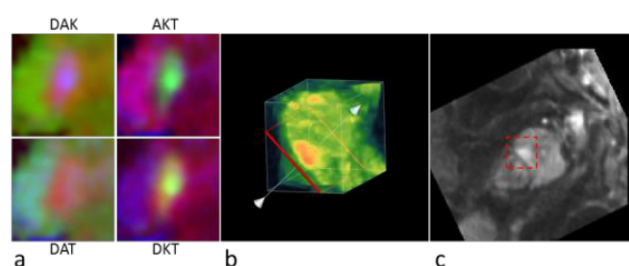
Tabuľka 1: Porovnanie prác diagnostiky klinicky signifikantnej RP

### 2.3.1 Publikácia s názvom *Prostate Cancer Diagnosis using Deep Learning with 3D Multiparametric MRI* [18]

V roku 2017 prišli autori článku *Prostate Cancer Diagnosis using Deep Learning with 3D Multiparametric MRI* s návrhom novej architektúry hlbokkej CNN s názvom XmasNet, ktorá slúži na klasifikáciu lézií rakoviny prostaty z 3D multiparametrických MRI dát. Riešenie pozostáva z troch častí a to predspracovanie dát, augmentácia dát a navrhnutá architektúra XmasNet siete.

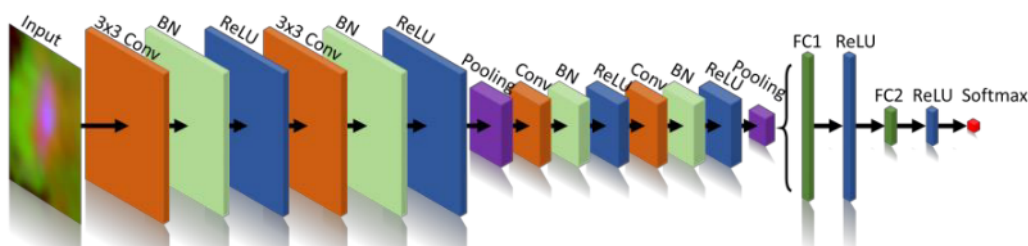
<sup>4</sup>Zdroj: <https://wiki.cancerimagingarchive.net>,

V predspracovaní boli upravované 3D MRI snímky rakoviny prostaty. Všetky snímky s rozličnými parametrami boli interpolované na rovnomerné rozlíšenie 1mm pomocou lineárnej interpolácie. Pre každý prípad prebehla registrácia rôznych typov MRI snímok. Po tom ako už boli všetky typy zobrazení na seba namapované, získali centrum lézie, ktoré vypočítali ako centroid tejto lézie. Región celej lézie segmentovali pomocou techniky *region growing* s využitím DWI. Pre trénovanie pripravili štyri druhy vstupov rôznym kombinovaním DWI (D), ADC (A), Ktrans (K) a traverzovaných T2W (T) snímok do RGB kanálov, ako môžete vidieť aj na obrázku 7.



Obr. 7: Architektúra XmasNet siete

Augmentácia dát bola robená pomocou 3D rotácie okolo centra lézie. Pod 7 rôznymi uhlami bol vyňatý jeden rez. Každý rez bol ešte v rovine náhodne otáčaný a posúvaný o jeden pixel. Na konci dosiahli 207144 trénovacích obrázkov, z ktorých každý zodpovedal regiónu veľkosti 32x32, ktorý obkolesuje centrum lézie. Týmto spôsobom sa podarilo preformulovať 3D problém na 2D problém.



Obr. 8: Predspracovanie a augmentácia dát pre XmasNet.

Na obrázku 8 môžeme vidieť architektúru siete XmasNet, ktorá pozostáva zo

4 konvolučných vrstiev, batch normalization (BN), ReLU, MaxPooling vrstiev a jednej plne prepojenej vrstvy ukončenou softmax funkciou. Pre finálnu predikciu použili ensembled metódu, kedy natrénovali viacero XmasNet modelov s rôznymi kombináciami vstupov a predikcia bola výsledkom váhovaného priemeru týchto modelov. Do výsledneho ensembled vybrali modely, ktoré pri tréningu v jednotlivých iteráciách maximalizovali validáciu AUC. Ich váhy sa rovnali počtu krát, kedy boli vybrané. Trénovanie zastavili, keď validácia skonvergovala.

Výsledný model obsahoval 7 XmasNet modelov, z ktorých päť dosahovalo výborné výsledky a taktiež boli pridané dva modely, ktoré dosahovali slabé výsledky, ale poskytovali komplementárne informácie voči ostatným. Podarilo sa im dosiahnuť úspešnosť 0.89 metódou AUC na testovacích dátach. Toto riešenie dosiahlo druhé najvyššie AUC vo výzve PROSTATEx 2017.

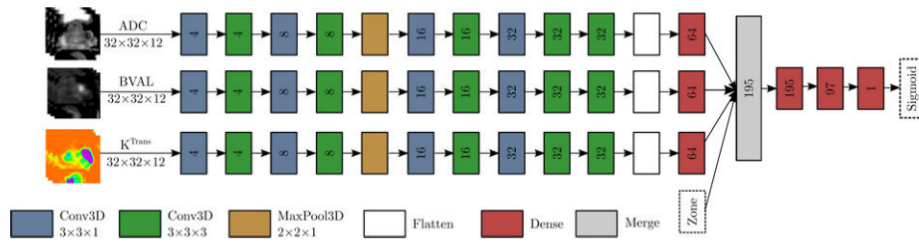
### **2.3.2 Publikácia s názvom *Classification of Clinical Significance of MRI Prostate Findings Using 3D Convolutional Neural Networks* [20]**

Kolégium autorov v článku *Classification of Clinical Significance of MRI Prostate Findings Using 3D Convolutional Neural Networks* z roku 2017 publikovali riešenie, ktoré je založené na 3D konvolučných neurónových sieťach, ktoré dostávajú na vstupe rozdielne typy MRI snímok.

Tak ako je zvykom pri algoritmoch hlbokého učenia, aj tu prebehlo predspracovanie a rozšírenie dát pred začatím tréningu. Potom ako boli dáta predspracované, prebehlo rozšírenie datasetu za pomoci otáčania a posúvania obrázkov. Takto vznikol dataset, ktorý obsahoval 10000 obrázkov pre každú z piatich podmnožín krížovej validácie 2000 obrázkov. Intenzita obrázkov bola normalizovaná na interval 0 až 1. Ako posledný krok v predspracovaní bola z 3D obrázku vyňatá vzorka, v ktorej strede bolo centrum nálezu. Pre T2W snímky sa vyňala 3D vzorka o veľkosti 40x40x40 mm a pre DWI a DCE snímky 32x32x12 mm.

Architektúra CNN siete zahŕňa 3 vstupné kanály pre ADC mapy a maximálne b-value z DWI, a Ktrans získaných z DCE MRI. Na prvej plne prepojenej vrstve je ešte pridaná komplementárna informácia o zóne, v ktorej sa nález nachádza. Na obrázku 9 môžete vidieť túto architektúru zloženú z deviatich konvolučných vrstiev, maxpooling vrstvy. Výstupy z posledných konvolučných vrstiev boli spojené do

jedného vektoru spolu so zonálnou informáciou nálezu.



Obr. 9: Architektúra 3D CNN, ktorá ma na vstupe ADC, b-value (BVAL) a Ktrans snímky spolu so zonálnou informáciou.

Pri tréňovaní využili algoritmus *stochastic gradient descent* spolu s Adam metódou aktualizácie, mini-batch s veľkosťou 64 a stratovou funkciou *binary cross-entropy*. Cross-validáciu použili na nájdenie najlepšej kombinácie vstupných kanálov a na nájdenie najlepšieho počtu filtrov pre konvolučné vrstvy. V tejto štúdii dosiahli porovnateľné výsledky s expertnými radiológmi. Najlepšiu úspešnosť sa podarila dosiahnuť na typoch snímok, ktoré boli prezentované pri architektúre siete. Táto úspešnosť bola vyhodnotená AUC na 0.80, čo bol priemer 4 najlepších modelov. Najlepší z týchto modelov dosiahol úspešnosť AUC 0.85. Práca odprezentovala efektívne využitie 3D CNN pri detekcii klinicky signifikantných nálezoch.

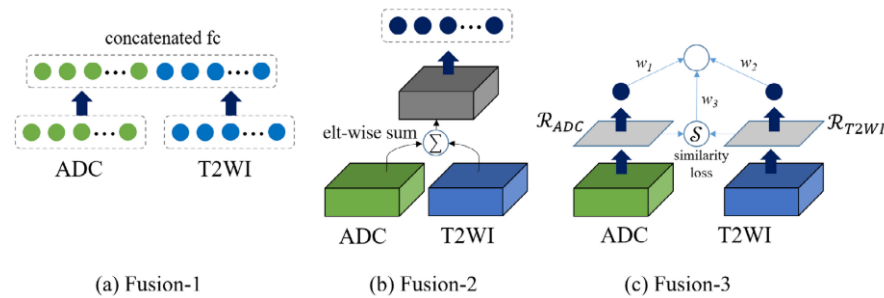
### 2.3.3 Publikácia s názvom *Automated diagnosis of prostate cancer in multi-parametric MRI based on multimodal convolutional neural networks*. [16]

V tejto práci sa sústredili na preskúmanie troch kritických faktorov v multimodálnych konvolučných sieťach pre diagnostiku rakoviny prostaty, ktoré ešte neboli podrobené hlbšej analýze.

Prvým faktorom je nedostatok dát v tejto oblasti a spôsoby ktorými sa tieto dáta augmentujú. Uvádzajú, že obvyklé metódy augmentácie, ako náhodné posunutie, rotácia či obracanie, ktoré sa široko využívajú pri spracovaní prirodzeného obrazu, nemusia byť úplne vhodné pre problémy medicínskych dát. Lézie RP sú sami o sebe *nerigidné* objekty, preto aplikovaním *nerigidnej* transformácie na originálne lézie rakoviny prostaty, by sme mali simulovať viac autentických prípadov lézií a zvýšiť tým rôznorodosť dát. V štúdií využili pri porovnaní tri augmentačné stratégie: (1)

Globálna rigidná transformácia zahŕňajúca posúvanie, otáčanie a obracanie. (2) Len nerigidná transformácia, (3) spojenie oboch rigidnej a nerigidnej transformácie.

Druhým bodom skúmania boli fúzia ADC a T2W príznakových vektorov v CNN. Využili dve známe metódy (a) zreťazenie príznakov a (b) elementárne súčty máp príznakov. Experimentálne výsledky ukazujú, že nelineárne korelácie medzi vysokorozmernými ADC a T2WI CNN príznakmi sa sotva môžu zachytiť jednoduchou stratégiou váženia. Metóda elementárnej sumácie príznakov (angl. Element-wise summation) dokáže zachytiť priestorovo konzistentné vizuálne vzory, vďaka čomu je možné vyťažiť viac relevantných vlastností RP. V tomto smere autori vytvorili (c) novú fúznú metódu, ktorá generuje mapy príznakov pre každú modalitu a každý pixel na mape označuje pravdepodobnosť, že tento pixel patrí do výstupnej triedy. Na obrázku 10 môžete vidieť všetky tri typy fúzií, ktoré boli v práci porovnávané.



Obr. 10: Ukážky troch prístupov fúzie príznakových vektorov.

Tretím faktorom, ktorý bol zahrnutý do analýzy je vplyv troch state-of-the-art architektúr konvolučných sietí s trocha rozličnými hĺbkami. V štúdiu porovnali VGGNet [24], GoogLeNet [26] a ResNet [10] architektúry multimodálnych CNN. Výsledky však ukázali, že rozdiely v úspešnosti sú medzi týmito troma architektúrami malé.

Aby sme sa dostali k finálnemu zhrnutiu výsledkov, musíme si obajsníť niektoré časti riešenia, ktoré sme ešte nespomenuli. Vstupom do siete je dvojica T2W a ADC snímok, ktorých intenzity boli predom normalizované a prebehla aj registrácia na zarovnanie ADC podľa zodpovedajúceho T2W. Zo snímok ADC za pomoci radiológov manuálne vyseletovali oblasť záujmu, kde sa nachádza lézia. Augmentácia prebehla prostredníctvom už vyššie spomínaných metód, kedy pomocou každej na-

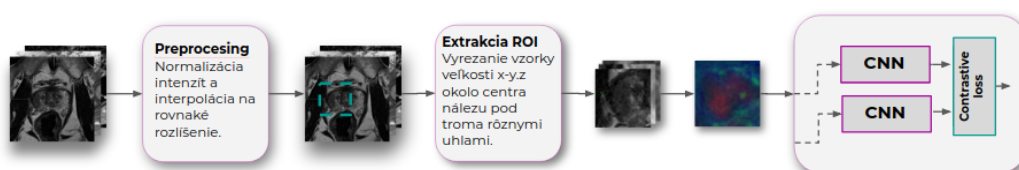
generovali približne 10000 dvojíc ADC-T2W. Architektúra siete obsahovala dve nezávisle multimodálne CNN, kde mala každá na vstupe jednu z modalít. Potom ako prebehla fúzia príznakov oboch CNN, bol model skombinovaný so state-of-the-art metódou [9] manuálnej extrakcie črt. Príznakové vektory oboch modelov boli vstupmi do SVM klasifikátora, ktorého výstup bol finálnou predikciou.

Vo výslednom experimente vyhodnotili porovnanie troch hore uvedených faktorov vplyvujúcich na úspešnosť predikcie. Z výsledkov pri porovnávaní augmentačných prístupov nad dátami vyplynulo, že tretia zo spomínaných metód, kedy skombinovali non-rigid a rigid transformáciu dokázala poskytnúť väčšiu rozmanitosť dát a dosahovala najlepšie skóre AUC na obidvoch modalitách na vstupe. Druhým pozorovaným faktorom boli metódy fúzie dvoch CNN kanálov podľa obrázka 10. Zistili, že najlepším prístupom bola ich novonavrhnutá metóda Fusion-3, ktorá dokáže najefektívnejšie spojiť komplementárne informácie o T2W a ADC. Fusion-2 je tak tiež lepšia ako prvá Fusion-1 metóda, pretože dokáže lepšie zachytiť informácie o priestorovej konzistentnosti T2W a ADC. V treťom bode pozorovania sa sústredili na úspešnosť rôznych architektúr multimodálnych CNN. Výsledky porovnávané metódou AUC ukázali, že výber architektúry CNN nemá na konečný výsledok zásadný vplyv. Napriek tomu najlepšia úspešnosť 0.9068 bola dosiahnutá na VGG-Net architektúre, zvyšné dve dosiahli v rozmedzí do 0.02 horšie výsledky. V poslednom kroku vyhodnotili spojenie s ručne extrahovanými príznakmi [9], ktoré samostatne dosahovalo  $AUC = 0.7342$ . Skombinovanie ručnej extrakcie s ResNet zvýšilo úspešnosť predpovede na 88.75% a  $AUC=0.9087$ .

## 3 Návrh

### 3.1 Koncept riešenia

Pri návrhu nášho riešenia sa zameriame na využitie počítačového videnia a hlbokého učenia pre diagnostiku klinicky významných lézií prostaty v mpMRI dátach. Architektúra nami navrhutej metódy je zobrazená na obrázku 11. Na vstupe nášho modelu sú 3D MRI dáta rôznych sekvencií (T2W, ADC, bval, Ktrans). Tieto dáta sú predspracované, kedy normalizujeme intenzity jednotlivých snímok, aby sme odstránili rozdiely v intenzitách medzi rôznymi MRI sekvenciami alebo aj v rámci jednej modality. Následne jednotlivé snímky interpolujeme na jednotné rozlíšenie. Ďalšou časťou našej metódy je extrakcia regiónu, v ktorom sa nachádza nález rakoviny. Pozorovaný región záujmu získavame z troch orientácií, aby sme získali viac ako len jeden pohľad na konkrétny nález. Výstupom doterajšieho predspracovania sú 2D obrázky obsahujúce detail konkrétnych nálezov. Každé jedno pozorovanie má viacero vzoriek z rôznych MRI sekvencií, ktoré medzi sebou kombinujeme do jedného viac-kanálového obrázka. Pre klasifikáciu a získavanie príznakov sme navrhli architektúru Siamských konvolučných neurónových sietí (SCNN), ktorá pozostáva z dvoch identických CNNs a Contrastive loss funkcie. Vstupom pre našu SCNN je pár dvoch obrázkov, ktoré sú označované podľa toho, či patria do rovnakej triedy. Finálna predikcia je získaná kombináciou predikcie troch SCNN. Každá natrénovaná pre jednu z troch vyššie spomenutých orientácií. V nasledujúcej časti si detailnejšie predstavíme jednotlivé moduly nášho riešenia.



Obr. 11: Ukážka extrakcie regiónu pozorovania, ktorý obsahuje klinicky významnú léziu prostaty.

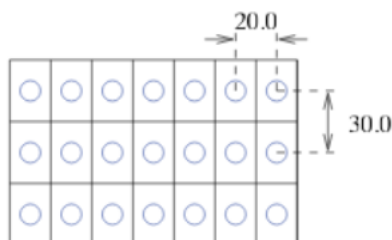


### 3.1.1 Predspracovanie

Obvyklou súčasťou metód, ktoré pri klasifikácii využívajú kombináciu viacerých MRI modalít je ich registrácia. Pomocou vzájomnej registrácie sa snímky rôznych MRI modalít navzájom zarovnávajú a odstránia nezrovnalosti spôsobené počas zaznamenávania snímok. Náš dataset obsahuje jednotlivé štúdie, v rámci ktorých sú už snímky registrované, takže registráciu už aplikovať nebudeme.

Jedným z krokov predspracovania dát je normalizácia, ktorú aplikujeme nad dátami, aby sme odstránili odlišnosti, ktoré mohli byť spôsobené napríklad odlišným nastavením zariadenia alebo použitím iného zariadenia na získanie snímok. Pre úpravu intenzít sme pripravili viacero normalizačných techník. Jednou zo základných prístupov, ktorý využívame je škálovanie intenzít do rozsahu 0-1. Ďalšou metódou, ktorá sa ponúka je štandardizácia dát pomocou Z-score normalizácie. Takto dosiahneme, že priemer intenzít bude 0 a ich štandardná odchýlka 1.

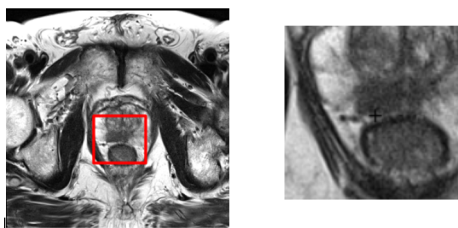
Ďalším spoločným problémom pri využívaní mpMRI sú rozdielne fyzické vzdialenosti pixelov medzi rôznymi MRI modalitami alebo aj v rámci jednej modality, čo je ovplyvnené iným nastavením snímania. Aby sme mohli spoločne kombinovať viacero MRI modalít musia byť vzdialenosti medzi pixelmi jednotné. Na obrázku 12 môžete vidieť dvojrozmernú mriežku pixelov, kde sa v každom smere vzdialenosti medzi pixelmi líšia. Naše dáta upravíme tak, aby mali v každom smere rovnaké vzdialenosti, čo znamená že v niektorých prípadoch treba mriežku s pixelmi zhustiť. Následkom toho vznikajú nové pixely, ktoré predtým neboli definované a musíme zistiť ich novú hodnotu. Hodnota je dopočítaná pomocou interpolácie medzi susednými pixelmi. Je možné použiť viacero metód interpolácie ako lineárna, bilineárna alebo kubická interpolácia.



Obr. 12: Dvojrozmerná mriežka s pixelmi a vzdialenosťmi medzi nimi.

### 3.1.2 Extrakcia regiónu záujmu (ROI)

Pre úspešnú klasifikáciu a extrakciu črt je nevyhnutné zamerať pozorovanie len na región, kde sa nález nachádza. V niektorých štúdiách pre získanie regiónu aplikovali segmentáciu nádoru, čím sa podarilo získať presnú oblasť zasiahnutú rakovinou. Druhým prístupom je extrahovanie regiónu určitej veľkosti okolo centra nálezu. V našej práci navrhujeme riešenie extrakcie regiónu obklopujúceho pozorovaný nádor, na základe informácie o pozícii nálezu, ktorá je súčasťou datasetu. Pre každé pozorovanie získame vzorku o veľkosti  $x*y*z$  pixelov, čiže je možné nadobudnúť 2D, ale aj 3D obrázok. Takto získame tri obrázky, každý nadobudnutý v jednom z troch rôznych uhlov. Na obrázku 13 sa naľavo nachádza rez 3D MRI obrázkom a pozorovaným bodom. Napravo je vyobrazený extrahovaný región, ktorý obsahuje klinický signifikantnú RP. Týmto spôsobom vytvoríme sadu tréovacích obrázkov.



Obr. 13: Ukážka extrakcie regiónu pozorovania, ktorý obsahuje klinický signifikantnú léziu prostaty.

### 3.1.3 Augmentácia

V tejto práci sa sústredíme na plné objavenie možností hlbokých SCNN modelov natrénovaných len na dátach dostupných v datasete. Z dôvodu, že náš dataset je nevyvážený (254 nezhubných a 76 zhubných) sme sa rozhodli vyskúšať dosah augmentácie na vybalancovanie nášho datasetu. Ako bolo demonštrované v štúdii [16], aplikovaním oboch rigidnej aj nerigidnej augmentácie môžu modely dosiahnuť najlepšiu úspešnosť. Rozhodli sme sa aplikovať rigidnú deformáciu zahŕňajúcu náhodné posunutie, otočenie, zrezanie a prevrátenie. Druhou navrhovanou deformáciou je elastická deformácia spolu s otočením a prevrátením obrázku.

Druhou metódou vybalancovania datasetu je podvzorkovanie dát, aby sa počet

vzoriek v triedach rovnal. Odstránenie prebieha až počas vyhodnocovania modelu našimi metrikami.

### 3.1.4 Model a klasifikácia

Pre úlohu klasifikácie sme navrhli Siamské neurónové siete, ktoré boli úspešne použité pri one-shot klasifikácii [14]. Aplikácia pre mediccké dáta bola skúmaná v štúdií [30], kde experimentálne výsledky ukázali, že SCNN je porovnateľná so state-of-the-art CNN a vyžaduje omnoho menej dát na trénovanie. V našej práci sa zameriame na využitie hlbkej SCNN pre úlohu rozlíšenia medzi zhubnou a nezhubnou RP.

Hlboká SCNN architektúra je variantou neurónovej siete, ktorá dokáže nájsť vzťahy a podobnosti medzi vstupnými objektami. Naša architektúra sa skladá z dvoch identických CNN podsietí, zdieľajúce rovnaké parametre a váhy, ktoré sa počas tréningu upravujú rovnako. Ako podsiete boli zvolené CNNs kvôli ich širokému využitiu v rámci spracovania obrazu a množstvu súčasných state-of-the-art riešení v oblasti diagnostiky rakoviny prostaty.

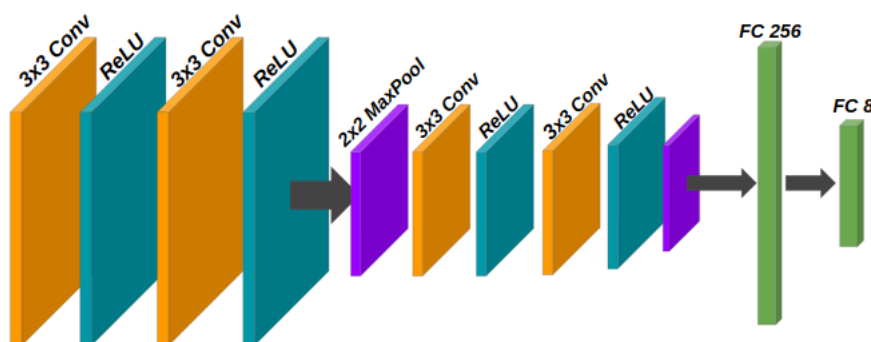
Vstupom do každej z podsietí môže byť viackanálový obrázok podobne ako RGB obrázky. Doterajším predspracovaním trojrozmerných MRI dát sme dostali zmenšené vzorky, ktoré zachytávajú len konkrétny nález a jeho blízke okolie. Pre každý nález získame sadu obrázkov v iných MRI modalitách. Tieto modalitty budeme kombinovať do jedného viackanálového obrázka, ktorý sa zhoduje so vstupom našej CNN podsiete. Vstupný obrázok môže byť kombinácia ľubovoľných MRI modalít, čo nám prinesie slobodu pri budúcich experimentoch a nájdení najlepšej kombinácie vstupov.

Vyššie sme si vysvetlili, aké vstupy očakávajú podsiete nášho SCNN modelu. V závislosti od toho, že náš model obsahuje dve CNN podsiete, bude aj model SCNN očakávať pár predspracovaných obrázkov. Vstupy budeme vytvárať kombináciou ľubovoľných dvoch nálezov, ktoré budú anotované podľa toho, či patria do rovnakej triedy. Ak obrázky patria do rovnakej triedy, čiže sú oba nálezy zhubné alebo oba nezhubné, priradíme im štítok 1. Ak sú oba z inej triedy, priradíme im štítok 0. Takýmto spôsobom dokážeme vytvoriť  $\binom{N}{2}$  jedinečných trénovacích vstupov.

Stratová funkcia, ktorú v našom modeli využijeme je Contrastive loss funkcia.

Táto funkcia dostáva ako vstup vektory príznačkov, ktoré sú výstupom CNN podsietí a štítok definujúci ich vzťah. Pomocou euklidovej vzdialenosti je vypočítaná ich podobnosť a funkcia sa snaží o to, aby podobné objekty boli združené spolu a rozdielne objekty vykazovali, čo najmenšiu podobnosť. Inými slovami povedané, Contrastive loss funkcia sa snaží odhaliť spoločné črty pre konkrétnu triedu a tie rozdielne pre objekty iných tried.

Pri návrhu a vhodnom výbere vrstiev našej CNN sme sa inšpirovali prácou [18], ktorá podobne ako my kombinuje rôzne MRI modalities do jedného viackanálového vstupu. Na obrázku 14 môžete vidieť zloženie jednotlivých vrstiev. Sieť sa skladá zo štyroch konvolučných vrstiev s ReLU aktivačnou funkciou a dvoch max pooling vrstiev. Sieť je zakončená dvoma plne prepojenými vrstvami.



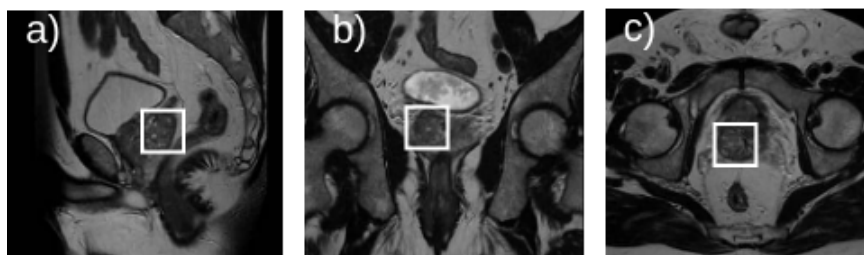
Obr. 14: Architektúra navrhnutej CNN.

### 3.1.5 Trénovanie modelu

V procese tréovania použijeme metódu 5-násobnej krížovej validácie, kedy naše predspracované dáta rozdelíme na 5 rovnakých častí a jednu použijeme ako validačnú vzorku a zvyšné poslúžia ako tréovacie dáta. Pri tréovaní využívame Adam optimalizačný algoritmus. Váhy v CNN inicializujeme náhodne pomocou Xavier metódy. Proces tréovania budeme kontrolovať pomocou grafov vykresľujúcich hodnoty loss funkcie a úspešnosť na validačných dátach pre dané iterácie. Z grafov vyčítame moment, kedy hodnoty validačnej vzorky prestanú konvergovať a finálny model natrénujeme s rovnakými parametrami, ale len s počtom iterácií v momente,

kedy loss funkcia skonvergovala.

Pomocou vyššie spomenutých metód natrénujeme tri SCNN modely, každý na sade obrázkov s inou orientáciou pozorovanej lézie. Finálna predikcia bude spočívať v kombinácii predikcií, z týchto troch rôznych pohľadov na jednu léziu.

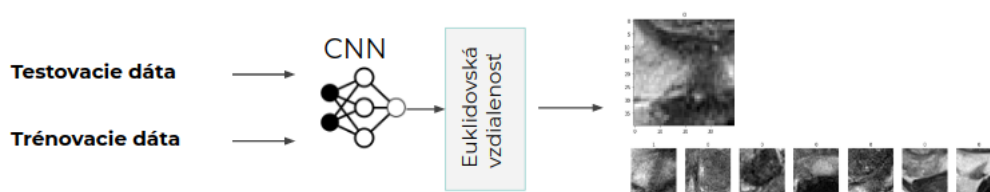


Obr. 15: Ukážka troch pohľadov na nález.

### 3.1.6 Metódy vyhodnocovania

Vyhodnocovanie nášho modelu sa líši od klasických CNN, kde výstupom siete je pravdepodobnosť príslušnosti do nejakej z tried. V našom prípade po natrénovaní modelu získavame samostatný CNN model s jeho váhami, pomocou ktorého získame vektor príznakov pre vstupný objekt. Takto získame vektory príznakov pre všetky tréningové aj testovacie dáta. Následne pre každý testovací objekt pomocou euklidovskej vzdialenosti vypočítame podobnosť so všetkými tréningovými dátami. Podľa podobnosti vieme zoradiť tréningové dáta od tých najpodobnejších k pozorovanému nálezu. V našej práci sme určili dve metódy, ktorými určíme, či sa jedná o zhubný alebo nezhubný nález:

1. Pozorovaný nález priradíme do triedy, do ktorej patrí najpodobnejší z tréningových dát.
2. Pozorovaný nález prehlásime za zhubný, ak sa v sade desiatich najpodobnejších objektov vyskytuje určité percento zhubných nálezov.



Obr. 16: Ukážka metrík vyhodnocovania.

## 3.2 Použité dáta

Podstatnou časťou návrhu sú aj dáta, ktoré sa budú v riešení využívať. Zdroj našich dát je inštitút *The cancer image archive*, ktorý uverejnil dataset s 3D MRI snímkami rakoviny prostaty. Tento dataset je taktiež súčasťou výzvy PROSTATEx, zameranú na metódy analýzy snímok pre diagnostiku klinicky významnej rakoviny prostaty.

Dataset je kolekciou MR štúdií prostaty. Každá štúdia zahŕňa T2-weighted (T2W), proton density-weighted (PD-W), dynamic contrast enhanced (DCE) a diffusion-weighted (DW) zobrazovanie. Uverejnená časť so štítkami pre tréning obsahuje štúdie 204 pacientov. Nadobudnuté 3D MR snímky prichádzajú vo formáte DICOM a každá štúdia je doplnená o Ktrans obrázky. Dataset obsahuje aj komplementárne informácie k jednotlivým štúdiám. Podstatnou informáciou je štítok, či sa jedná o klinicky významnú RP alebo nie. Ďalšou podstatnou informáciou sú koordináty pozorovaného miesta záujmu, kde sa lézia na snímke nachádza.



## 4 Implementácia

---

V tejto časti si opíšeme implementované moduly, ktoré sú využívané v jednotlivých fázach predspracovania, tréovania a vyhodnocovania. Popíšeme funkčnosť, spôsob ich implementácie a knižnice, ktoré sme využili.

### 4.1 Vývojové prostriedky

Implementácia bude riešená v jazyku Python. Využijeme prostredie Jupyter notebook, ktoré ponúka rýchlu dokumentáciu spustiteľného kódu a vizualizácie. Pomocou Jupyter notebooku budeme analyzovať dostupné dáta, budeme vykonávať experimenty a ich kontrolu. Výstupom bude dokumentácia podporená časťami kódu. Overené časti našej metódy prepíšeme do spustiteľných python skriptov.

V časti predspracovania využijeme python knižnicu *SimpleITK*, ktorá je určená na spracovanie medicínskych dát. Pomocou tejto knižnice budeme načítavať MRI dáta, vykonávať interpoláciu a transformácie. Ďalšími knižnicami využitými pri spracovaní dát sú *numpy*, *matplotlib*, *scikit-learn*. Pri implementácii modelov hlbokých neurónových sietí použijeme knižnicu *Tensorflow*.

### 4.2 Predspracovanie

V časti predspracovania upravujeme dáta z nášho datasetu na vhodný tvar pre neskoršie spracovanie modelmi hlbokých neurónových sietí. Surové dáta sú uložené v jednom súbore, ktorý je rozdelený na podadresáre konkrétnych pacientov. Pri každom z pacientov sú prítomné viaceré MRI štúdie, každá uložená pod špecifickým menom. Tieto MRI štúdie, ktoré sú 3D MRI snímky prečítame pre každého pacienta a upravené ich uložíme do nového adresára, ktorý presne definuje, o aký typ snímku sa jedná, orientáciu a akej je veľkosti. Příklad: originálne dáta */PROSTATEx/ProstateX-0000/07-07-2011-MR prostata kanker detectie WDSmc MCAPRODETW-05711/4-t2tsetra-00702/*, upravené dáta */data/t2tsetra/t/40x40x1/*.

Nasledujúci pseudokód popisuje postup krokov predspracovania našich dát. Vstupom je tabuľka *zoznamNálezov*, ktorá v jednotlivých záznamoch obsahuje meno pacienta, fyzické koordináty nálezu, zonálnu informáciu prostaty a štítok.



---

**Algoritmus 1** Predspracovanie

---

**Vstupy:** *zoznamNálezov*

**Výstupy:** *súborNálezov*

```
1: procedure PREPROCESS(zoznamNálezov, MRImodalita, veľkosť, orientácia, rozlíšenie)
2:   for each(zoznamNálezov) do
3:     cesta  $\leftarrow$  getPathToImage(nález.MenoPacienta, MRImodalita)
4:     MRIsnímok3D  $\leftarrow$  readImageFromPath(cesta)
5:     normaliser.fit(MRIsnímok3D)
6:     MRIsnímok3D  $\leftarrow$  resampleSpacing(MRIsnímok3D, rozlíšenie, interpolátor)
7:     vzoroka  $\leftarrow$  getPatch(MRIsnímok3D, nález.centrum, veľkosť, orientácia)
8:     vzoroka  $\leftarrow$  normaliser.Normalise(vzoroka)
9:     saveAsImage(vzoroka, nováCesta,)
10:   end for
11:   saveInformationsAboutImages(nováCesta)
12: end procedure
```

---

Pre vykonanie algoritmu sme vytvorili spustiteľný skript v jazyku python. Predtým ako tento skript spustíme, musíme definovať *MRImodalita*. Je to string definujúci, akú MRI modalitu chceme spracovať. *Veľkosť* je vektor troch čísel určujúcich rozmery finálneho obrázka. *Rozlíšenie* je taktiež vektor troch čísel, ktoré určujú veľkosti medzi pixelmi v mm v každom smere. Posledným údajom je *orientácia*, ktorá určuje ako má byť finálny obrázok orientovaný.

Súbor *zoznamNálezov* je súčasťou nášho datasetu a je vo formáte .csv. Tento súbor načítavame pomocou knižnice *Pandas*, ktorá nám ponúka aj prácu s tabuľkou pomocou dopytov nad ňou. Zo všetkých nálezov odstránime tie, ktoré nie sú vhodné pre neskoršie spracovanie. Následne iterujeme cez vyčistený *zoznamNálezov* a nad každým záznamom vykonáme sadu úloh:

- **getPathToImage** je funkcia, ktorá vráti cestu k MRI sérii zadanej modality pre konkrétneho pacienta. Pomocou knižnice *os* sa prelistujeme adresármi z koreňového adresára k adresáru, kde sú uložené všetky súbory s dátami pre daného pacienta. Pomocou regex vyfiltrujeme názov súboru, ktorý v názve obsahuje zadanú modalitu. Cestu k tomuto súboru funkcia vráti.
- **readImageFromPath** je funkcia, ktorá pomocou knižnice *SimpleITK* prečíta sériu MRI snímok v DICOM formáte a vráti 3D obrázok. Spolu s obrázkom

sú načítané aj metadáta, ktoré slúžia na neskoršie spracovanie obrázku a sú potrebné pri priestorovej orientácii v obrázku.

- **normalizer** je objekt, ktorý slúži na normalizáciu našich dát. Vytvorili sme sadu normalizačných metód ako MinMax, Zscore normalizáciu a Gausovú normalizáciu. Pri spustení predspracovania sa môžeme rozhodnúť, ktorú z metód použijeme. Pre každú z metód je vytvorený objekt, ktorý sa dá použiť ako *normalizer*. Následne sú volané jeho metódy *fit()* a *normalise()*. Metóda *fit()* dostane na vstupe 3-rozmerný *numpy.array* pixelov obrázku, z ktorých získa atribúty potrebné pre normalizáciu. Následným zavolaním metódy *normalise()* sa vykoná operácia normalizácie nad vstupnými dátami a je vrátený upravený *numpy.array*.
- **resampleSpacing** je funkcia, ktorá má na vstupe *MRI snímok 3D*, rozlíšenie a interpolátor. Výstupom je obrázok s rozlíšením, aké sme zadali na vstupe. V tejto funkcii využijeme knižnicu *SimpleITK* a jej filter *ResampleImageFilter()*. Pomocou tohto filtra dokážeme zmeniť rozlíšenie obrázku, čiže vieme zmeniť počet pixelov v každom smere. Filter vyžaduje definovanie interpolátora, ktorý ponúka knižnica *SimpleITK*. Základnými príkladmi sú *sitkNearestNeighbor*, *sitkLinear*, *sitkBSpline* a ďalšie, ktoré posielame do funkcie cez argument *interpolátor*. Filter taktiež vyžaduje definovanie novej veľkosti obrázku, ktorú musíme vypočítať na základe nami novo zvoleného rozlíšenia takto:

$$newSize = \frac{originalSpacing * (originalSize - 1)}{newSpacing} + 1$$

- **getPatch** využívame pri extrakcii regiónu záujmu (ROI). Vstupom funkcie je *MRI snímok 3D*, *nález.centrum*, *veľkosť*, *orientácia* a výstupom je obrázok v zadanej veľkosti a orientácii. Môžeme si vybrať jednu z orientácií, pod akou chceme obrázok nadobudnúť. Tieto orientácie sú *t*, *s*, *a*, pri predstave 3D obrázka ako kváder je to pohľad zhora, spredu a zboku. Namiesto rotovania obrázku, zmeníme len tvar vektora veľkosti nasledovne:

---

```
if orientacia == 's':
    velkost = (velkost[0], velkost[2], velkost[1])
elif orientacia == 'a':
```

```
velkost = (velkost[2],velkost[1],velkost[0])
```

---

Nasledovne pomocou centra a veľkosti vypočítame hranice nového obrázku. Overíme, či hranice nepresahujú hranice originálneho obrázku. Ak sú hranice v poriadku, vyjmeme pixely z pôvodného, do nového *numpy.array* v požadovanom tvare. V prípade, ak bol obrázok nadobudnutý v inej orientácii ako pohľad zhora, teda zatiaľ nadobudnutý tvar obrázka v prípade orientácie *s* je napríklad (1x40x40), ale my požadujeme obrázok tvaru (40x40x1) pixelov. Pretransformujeme pixely do tohto tvaru a takýto obrázok funkcia vráti.

- **saveASImage** a **saveInformationsAboutImages** ako prvá spomenutá funkcia ukladá obrázok do súboru s jeho novým originálnym tvarom v *.nii* formáte. Druhá spomenutá funkcia uloží komplementárne informácie pre súbor vytvorených obrázkov s ich štítkami v *.csv* formáte.

### 4.3 Trénovanie modelu

V tejto časti sa venujeme trénovaniu modelu pomocou predpripravených dát. Predtým, ako začneme proces trénovania modelu, definujeme niektoré nevyhnutné vstupy. Určíme, aké modalities využijeme pri natrénovaní modelu. Modalita je definovaná ako cesta k adresáru, kde sa obrázky nachádzajú. Ďalej zadáme počet epôch trénovania, typ modelu SCNN, veľkosť *batchSize* a definujeme, či má byť využitá augmentácia. Pri voľbe augmentácie sa môžeme rozhodnúť medzi viacerými augmentačnými technikami. Po natrénovaní modelu sa ukladajú optimalizované váhy a parametre do externého súboru.

---

**Algoritmus 2** Trénovanie

---

```
1: procedure TRAINING(modalities, batchSize, epoch)
2:   loader  $\leftarrow$  DataLoader()
3:   loader.loadData(modalities)
4:   if len(modalities)  $\neq$  1 then
5:     loader.combineChanels(modalities)
6:   end if
7:   model  $\leftarrow$  AnyModel()
8:   for each(loader.generateKfolds()) do
9:     dataset  $\leftarrow$  Dataset()
10:    dataset.setData(Kfold.data)
11:    if augmentation then
12:      augmentor  $\leftarrow$  AnyAugmentor()
13:    end if
14:    for each(epoch) do
15:      batch  $\leftarrow$  dataset.getSiameseBatch(batchSize, augmentor)
16:      model.train(batch)
17:      PlotTrainingProcess(model)
18:    end for
19:    model.evaluate(dataset.testData)
20:    model.writeParameters()
21:  end for
22: end procedure
```

---

Pre každý natrénovaný model vytvárame záznam s informáciami o použitom modeli, o dátach na ktorých je natrénovaný a nastavených parametroch optimalizácie. Tieto záznamy sa ukladajú do externého súboru, ktorý používame neskôr pri analýzach a znovupoužití modelov. Pri vytváraní a zapisovaní záznamu využívame knižnicu *Pandas*.

- **Augmentácia** súbor objektov slúžiacich pri augmentácii dát. Využívame knižnice *Albumentations* a *Augmentor*, ktoré obsahujú množiny transformačných funkcií, ktoré spolu kombinujeme, aby sme dosiahli nami požadovanú úroveň deformácie vstupných dát. Vytvorili sme tri druhy augmentácie: rigidná, nerigidná a ich kombinácia. Každá je popísaná jedným objektom, ktorý má sadu metód. Tieto metódy využívame pri znásobení vstupných dát alebo deformáciu len jedného obrázku.
- **DataLoader** je objekt, ktorý pomáha pri načítavaní a správe tréningových dát.

Základnou funkciou, ktorú ponúka je načítavanie obrázkov, na čo využíva knižnicu *SimpleITK*. Najprv však zo zadaného adresára prečíta súbor s informáciami o dátach, kde nájde názvy súborov pre nálezy a podľa toho načíta obrázky. Obrázky sú transformované na viac dimenzionálny *numpy.array* a ich tvar je upravený na požadovaný vstup SCNN modelu.

Ďalšou zásadnou funkciou je spájanie viacerých modalít do jedného viac-kanálového obrazu. Objekt ukladá dáta pre každú modalitu ako *numpy.array*s. Pri ich spájaní využívame funkciu v tvare: *numpy.concatenate([modality1, modality2,...], axis=3)*. Funkcia uloží novo vytvorenú sadu obrázkov k ostatným datasetom pod menom "combined".

Poslednými funkciami sú tie, ktoré plnia úlohu rozdeľovania na testovacie a trénovacie dáta. Objekt typu *DataLoader* môže vrátiť rozdelený dataset na trénovaciu a testovaciu množinu, na čo využíva knižnicu *Scikit-learn* a jej funkciu *train\_test\_split()*. Objekt dokáže vrátiť aj generátor pre viacnásobnú krížovú validáciu. Generátor je implementovaný pomocou knižnice *Scikit-learn*.

- **Dataset** je objekt, ktorý slúži na vytváranie siamských párov a ich štítkov. Objekt je inicializovaný s trénovacími a testovacími dátami, ktoré vytvára objekt *DataLoader*. Jeho metóda *get\_siamese\_batch* náhodne vytvára kombinácie podobných alebo rozdielnych obrázkov. Využíva k tomu sadu pomocných funkcií využívajúcich *numpy* objekty.
- **Model** definuje náš model SCNN, ktorý je zložený z dvoch CNNs. Modely implementujeme v knižnici *tensorflow*, *tensorboard*. SCNN vytvárame skladaním *tensorflow* grafu z jednotlivých častí. Definujeme *tensorflow.placeholder* pre vstupné dáta a výstup modelu, optimalizačné funkcie a loss funkciu. CNN modely vyskladáme z vrstiev, ktoré ponúka *tensorflow* knižnica.

## 4.4 Vyhodnocovanie

Vyhodnotenie prebieha v samostatnej časti po natrénovaní modelu. Ako sme spomenuli vyššie, vytvárame záznamy o modeloch, ktoré sú uložené v externom súbore. Pomocou tohto súboru môžeme natrénované modely obnoviť a lepšie ich

vyhodnocovať. Na základe záznamu obnovíme model a načítame jeho testovacie a trénovacie dáta. Nad dátami vykonávame klasifikáciu metódami, ktoré sme opísali v stati 3.1.6 Výsledkom je vektor predikovaných hodnôt a vektor pravdivých hodnôt (0,1).

Pomocou knižnice *Scikit-learn* a jej metrických funkcií vyhodnocujeme úspešnosť modelov. Vyhodnotenia sú zapísané k záznamu modelu pre neskoršiu analýzu.

Modely vyhodnocujeme aj vizuálnou kontrolou snímkov. Pri zobrazovaní snímkov využívame knižnicu *matplotlib*. Implementovali sme viacero pomocných funkcií, ktoré zobrazujú snímky nálezov v kontexte vyhodnotenia predikcie alebo vzťahu predikovaných tried. S vektormi predikovaných štítkov pracujeme pomocou knižnice *numpy*.



## 5 Vyhodnotenie

---

V časti vyhodnotenia si rozpíšeme, aké experimenty a hypotézy sme si stanovili a metriky, ktorými budeme overovať tieto hypotézy. Zhodnotíme, aké výsledky dosiahli naše experimenty, a či boli naše hypotézy pravdivé.

### 5.1 Experimenty

Koncept nášho návrhu založeného na architektúre Siamských sietí predpokladá, že model sa bude schopný naučiť reprezentáciu dát, na základe len dostupných dát z nášho datasetu. Základným experimentom je overenie, či natrénovaný model SCNN dokáže úspešne rozlišovať medzi zhubným a nezhubným nádorom. Taktiež či je model porovnateľný so state-of-the-art riešeniami, založenými na samostatnej konvolučnej neurónovej sieti. Naším experimentom chceme preukázať, že je možné použiť architektúru Siamských sietí aj pri probléme diagnostiky RP, kde je prítomné len malé množstvo anotovaných dát. Metódy hlbokého učenia vyžadujú tisícky až desať-tisíce anotovaných dát potrebné k natrénovaniu modelu. Využitie takýchto modelov pri riešení nášho problému si vyžaduje použitie robustných augmentačných techník, ktoré umelo rozšíria dataset. Informácia v novovytvorených dátach nemusí úplne dôveryhodne simulovať nové prípady RP, čo môže mať za následok nesprávne natrénovaný model, a tým aj neúspešnú predikciu.

Základný experiment, ktorý sme popísali vyššie doplníme o ďalšie experimenty. Pomocou komplementárnych experimentov sa budeme snažiť nájsť najlepšie parametre a vstupy nášho modelu, aby sme dosiahli čo najlepšiu úspešnosť.

- **Modality MRI** popisujú tkanivo RP. Každá štúdia nálezu obsahuje viacero MRI modalít, každá dokáže popísať inú charakteristiku RP. Využitím viacerých modalít pri predikcii by sme mali dosiahnuť lepšiu úspešnosť. V experimente sa zameriame na analyzovanie jednotlivých modalít a ich príspevkom k lepšej predikcii. Náš model sa pokúsime natrénovať na jednotlivých modalitách a taktiež budeme kombinovať tieto modality do viackanálového vstupu. Výstupom experimentu bude nájdená najvhodnejšia kombinácia modalít, pri ktorých dosiahol model najlepšiu úspešnosť.



- **Tri orientácie** pohľadov na pozorovaný nález. V tomto experimente vykonávame predikciu jedného nálezu, nie pomocou jedného obrázku ale troch, každý z iného pohľadu na pozorovaný nález. Predpokladáme, že spojením predikcie troch modelov zvýšime presnosť predikcie. Každý z modelov je natrénovaný na jednej z troch orienácií nálezu. Dáta každej orientácie musia mať rovnaké parametre, čo sa týka voľby modalít, veľkosti obrázkov a ostatné parametre, ktoré nastavujeme pri predspracovaní dát.
- **Augmentácia** je spôsob, ktorým sa vysporiadávame s nedostatkom dát. V jednom z našich experimentov sa pokúsime overiť, či bude mať augmentácia vplyv na náš model, ktorý nevyhnutne nepotrebuje zväčšovať množstvo trénovacích dát. Ďalším dôvodom pre využitie augmentácie je nevyvážený dataset, kde prevažuje pomer jednej triedy voči druhej približne 70:30.

## 5.2 Metriky

Pre správnu optimalizáciu modelov potrebujeme metriky, ktoré vyhodnotia model. Pri overovaní využívame metriku oblasti pod krivkou (AUC) z ROI (angl. receiver operating characteristic). Presnosť, špecificita a senzitivita klasifikácie je definovaná ako:

$$Presnosť = \frac{(TP + TN)}{(TP + FP + FN + TN)}$$

$$Specificita = \frac{TN}{TN + FP}$$

$$Senzitivita = \frac{TP}{TP + FN}$$

kde TP, FP, TN a FN sú správne pozitívne, nesprávne pozitívne, správne negatívne a nesprávne negatívne.

Pri vyhodnocovaní sú tieto metriky dôležité, pretože chceme získať model, ktorý bude správne určovať obidve triedy. Dosiahnutie vysokej senzitivity je dôležité vzhľadom na predikciu zhubných nádorov RP. Snažíme sa natrénovať model, ktorý nesprávne odhadne čo najmenej zhubných prípadov RP, o čom hovorí vysoké skóre senzitivity a v oblasti medicíny je veľmi podstatné. Na druhej strane špecifickosť testu hovorí o tom ako úspešne sa nám podarilo nezaradiť negatívne nálezy k pozitívnym.

Zjednodušené povedané, aby pacienti s nezhubným nádorom neboli označení ako pozitívni. Špecifickosť a senzitivita nám dokážu v celku správne napovedať, ako je náš model úspešný. Samozrejme sa snažíme dosiahnuť to, aby obidve hodnoty boli čo najvyššie. ROC slúži na popísanie úspešnosti modelu, vzhľadom na obe metriky špecifickosti a senzitivity. Oblasť pod ROC krivkou AUC score sa stalo všeobecne zaužívanou technikou vyhodnotenia úspešnosti v tejto oblasti. Pri optimalizovaní modelov sa sústreďíme na skóre AUC, ktoré dosahujú tieto modely.

Vyhodnocovanie modelov pozostáva aj z rozdelenia dát na tréningovú, validačnú a testovaciu sadu. V našom prípade, keď dataset pozostáva len zo stoviek dát, sme za najvhodnejší spôsob validácie a vyhodnocovania zvolili viacnásobnú krížovú validáciu. Naše dáta rozdelíme na 5 rovnakých častí, ktoré slúžia na 5-násobnú krížovú validáciu našich modelov. Výslednú úspešnosť budeme počítať ako priemer všetkých piatich validácií.

Jednou z metód ako môžeme skontrolovať náš natrénovaný model je aj vizuálna kontrola. Prostredníctvom obrázkov dokážeme skontrolovať ako sa modelu darilo rozlišovať medzi triedami. Pri väčšine úloh prirodzeného obrazu je takáto kontrola zväčša jednoduchá, pretože ak vidíme na obrázku auto, očakávame od modelu predikciu, že na obrázku sa nachádza auto. Pri medicínskych dátach je zložitejšie rozpoznať, čo sa na obrázku nachádza, aj pre skúseného radiológa. Napriek tomu môžeme rozpoznať niektoré vlastnosti obrazu, ktoré majú vplyv na rozhodovanie nášho modelu.

Navrhli sme zopár pomocných funkcií, ktoré vykresľujú obrázky na základe vzťahu k predikcii. Prvou takou funkciou je vykreslenie množiny obrázkov, ktoré patria do jednej zo sady TP, FP, TN a FN predikovaných hodnôt. Takou metódou môžeme skontrolovať, ktoré obrázky sa modelu podarili zaradiť do správnej triedy, a ktoré naopak nie. Čítaním takýchto obrázkov môžeme odhaliť chyby, ktoré sa model naučil pre danú triedu.

Druhou funkciou je vykreslenie N najpodobnejších obrázkov z tréningovej sady k obrázku, ktorý sa snažíme predikovať. Funkcia slúži na vykreslenie vzťahu. Môžeme si na základe nej skontrolovať, či sú tieto najpodobnejšie obrázky k danému obrázku skutočne podobné a majú nejaké spoločné črty.

## 5.3 Výsledky

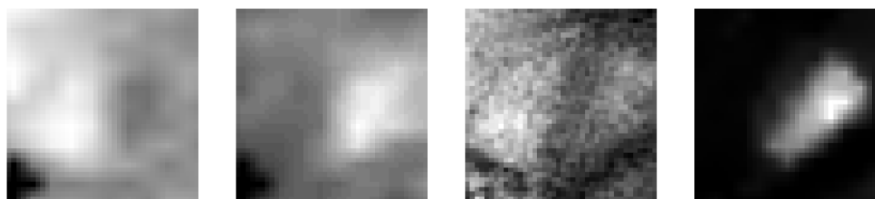
V tejto časti si popíšeme výsledky našich experimentov, pomocou vyššie uvedených metrík. Podarilo sa nám zostaviť model SCNN, ktorý sme trénovali na dostupných dátach. Náš model dokázal správne predpovedať triedu nálezu zo vstupných obrázkov. Nášmu modelu sa nepodarilo prekonať súčasné state-of-the-art riešenia, čo je spôsobené hlavne masívnou úrovňou predspracovania, ktorá je v týchto prácach publikovaná. Ďalším dôvodom je len obmedzený dataset v porovnaní s inými prácami, ktoré často pracujú so súkromnými dátami. Napriek tomu náš prístup vykázal lepšie výsledky s porovnaním samostanej CNN natrénovanej na našich dátach. Model sme podrobili nasledujúcim experimentom.

### 5.3.1 Experiment 1

V tomto experimente sme sa snažili analyzovať modalitu MRI a ich vplyv na úspešnú klasifikáciu. Pri našich experimentoch sme pozorovali ADC, T2W, bval a ktrans modalitu. Dáta boli upravené na jednotné izochromatické rozlíšenie 1mm. Veľkosť extrahovaných obrázkov je 20x20x1 pixelov, čo znamená, že využívame 2D obrázky. Dáta boli taktiež normalizované metódou MinMax do rozmedzia 0-1.

Pre optimalizáciu modelu bol využitý Adam optimalizér, kedy sme sa snažili nájsť najvhodnejšie parametre rýchlosti učenia a veľkosť dávky. Keď stratová hodnota správne klesala, pozorovali sme, v ktorej iterácii ešte stratová hodnota na tréningových aj testovacích dátach klesá. V momente, keď stratová hodnota na testovacích dátach prestala klesať, tento počet iterácií sme použili ako optimálny pre následné tréningy. Finálne vyhodnotenie sme získali s predom určenými parametrami tréningu, pomocou 5-násobnej krížovej validácie.

V tabuľke 2 môžete vidieť dosiahnuté úspešnosti jednotlivých modalít. Najvhodnejšou modalitou je ADC, za ktorou nasledujú ostatné. Na obrázku 12 môžete vidieť príklady zhubného nádoru, zobrazeného pomocou použitých MRI modalít. Naša sieť sa musí vysporiadať s nepresným regiónom lézie, keďže nevysegmentujeme presnú oblasť, ale lézia môže presahovať hranice obrázku alebo môže byť výrazne menšia ako nadobudnutý obrázok. Obrázok 12 ukazuje vlastnosti Ktrans modality, ktorá by mohla pomôcť modelu rozlíšiť, kde sa nádor nachádza.

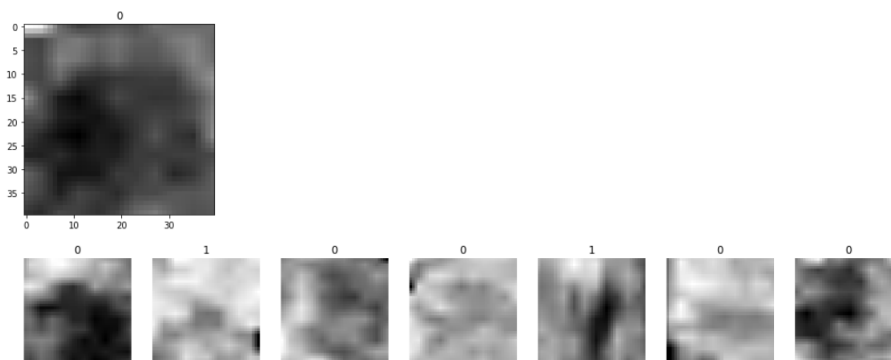


Obr. 17: Obrázky MRI zhubného nádoru zľava Bval, ADC, T2W a Ktrans.

MRI modalita	Presnosť	AUC skóre
Bval	0.6687	0.5673
T2W	0.7057	0.5898
Ktrans	0.6823	0.6177
ADC	0.7194	0.6488

Tabuľka 2: Úspešnosť modelov pri rôznych modalitách MRI

Jednu z našich vizuálnych metrík pozorovania modelu môžete vidieť na obrázku 18. Môžeme pozorovať, že v tomto prípade sú si najpodobnejšie obrázky skutočne podobné. Ostatné prípady môžeme usudzovať, že model sa naučil niektoré vzory, ktoré sú nám nie úplne viditeľné.



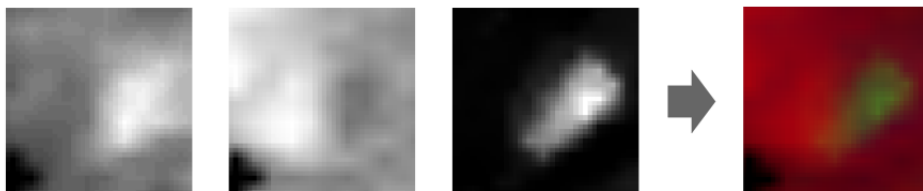
Obr. 18: Ukážka najpodobnejších nálezov z trénovacej sady k testovaciemu obrázku na ADC modalite.

Po experimentoch na samotných MRI modalitách sme sa posunuli k trénovaniu modelov s využitím kombinácie MRI modalít. Z výsledkov sme spozorovali, že aj

pri kombinácii modalít si udržujú vysoké skóre predikcie, kde sú prítomné modalita ADC a Ktrans, ktoré mali aj samotné najlepšiu úspešnosť. Úspešnosť predikcie nie je ovplyvnená len počtom skombinovaných modalít, ale hlavne výberom vhodných modalít. Z výsledkov však môžeme čítať, že kombinácia viacerých modalít zvyšuje úspešnosť predikcie. Kombináciou Ktrans a ADC sa nám podarilo zvýšiť AUC skóre o 4% oproti použitiu len samotnej ADC modality. Súhrn úspešnosti niektorých modelov môžete vidieť v tabuľke 3 a ukážku vstupu pre jeden z modelov v obrázku 19.

MRI modalita	Presnosť	AUC skóre
ADC+T2W	0.6684	0.5884
ADC+Ktrans	0.7278	0.6267
ADC+Ktrans+T2W+Bval	0.7196	0.6389
ADC+Bval+T2W	0.7196	0.6481
ADC+Bval+Ktrans	0.7231	0.6700
Ktrans+ADC	0.7184	0.6826

Tabuľka 3: Úspešnosti modelov pri kombinácií modalít MRI



Obr. 19: Ukážka kombinácie ADC, Bval a Ktrans do RGB obrázku.

### 5.3.2 Experiment 2

V tomto experimente sa zameriavame na predikciu jedného nálezu z troch rozdielných pohľadov: zhora, zboku a spredu. V tomto experimente sme museli vylúčiť Ktrans modalitu, pretože sme nedokázali získať snímky z troch uhlov, aby sme výrazne neovplyvnili veľkosť datasetu. Predikciu robíme pomocou kombinácie ADC,

Bval a T2W. V tabuľke 4 môžeme vidieť úspešnosť modelov natrénovaných s iným pohľadom na nález. V poslednom riadku je predikcia kombinácie týchto troch modelov, pomocou ktorej sa nám podarilo zvýšiť AUC skóre o viac než 2 % na 0.7190.

MRI modalita	Orientácia	Presnosť	AUC skóre
ADC+Bval+T2W	t	0.7196	0.6481
ADC+Bval+T2W	s	0.7363	0.6969
ADC+Bval+T2W	a	0.7532	0.6925
ADC+Bval+T2W	kombinované	0.7635	0.7190

*Tabuľka 4: Predikcia nálezu s tromi orientáciami*

### 5.3.3 Experiment 3

V poslednom experimente sme sa zamerali na problém nevyváženého datasetu a vplyv augmentácie. Použili sme rigidné a nerigidné techniky augmentácie dát, aby sme zistili, či má augmentácia vplyv aj na náš model SCNN, ktorý si vystačí aj s malým množstvom dát.

Z tabuľky 5 môžeme čítať úspešností jednotlivých modelov. Zistili sme, že pomocou augmentácie dát dokážeme zvýšiť úspešnosť predikcie. Pomocou rigidnej augmentácie sme dosiahli len zanedbateľné zlepšenie, ale spojením elastickej deformácie spolu s otáčaním a preklopením obrázka, sme zlepšili AUC skóre modelu o približne 3%.

V experimente sme testovali aj podvzorkovanie trénovacích dát, aby sme dosiahli vyváženú trénovaciu sadu. Podvzorkovaním sa nám nepodarilo dosiahnuť lepšie skóre ako pri nevyvážených dátach.

MRI modalita	Augmentácia	Presnosť	AUC skóre
ADC+Bval+Ktrans	bez	0.7231	0.6700
ADC+Bval+Ktrans	rigidná	0.7096	0.6768
ADC+Bval+Ktrans	rigidná+nerigidná	0.6992	0.6983
ADC+Bval+Ktrans	podvzorkovanie	0.6010	0.6441

*Tabuľka 5: Predikcia nálezu s tromi orientáciami*



## 6 Závěrečné zhodnotenie

---

V našej práci sa nám podarilo navrhnúť metódu hlbokých Siamských neurónových sietí pre úlohu automatickej klasifikácie lézií rakoviny prostaty s využitím 3D mpMRI dát. Naša metóda využíva výhody hlbokých konvolučných sietí, ktoré sa automaticky naučia reprezentáciu medických snímok a výhodu Siamských neurónových sietí, ktoré potrebujú len malý dataset na trénovanie. Naš SCNN model sa skladá z dvoch identických CNN sietí, ktorých architektúra je inšpirovaná štúdiou, ktorá je založená na podobnom prístupe kombinácie modalít.

Pre nájdenie najoptimálnejšieho vstupu pre model sme analyzovali ADC, T2W, Ktrans a Bval modalitu MRI a ich vplyv na úspešnú klasifikáciu. V prvom experimente sme sa zamerali na analyzovanie jednotlivých modalít. Z výsledkov vyplynulo, že najvhodnejšou pre náš SCNN model je ADC a po nej nasleduje Ktrans modalita. Pre zlepšenie úspešnosti nášho modelu sme pozorovali kombinácie jednotlivých MRI modalít do viackanálového vstupu. Z pozorovaní sme zistili, že vhodná kombinácia je ovplyvnená prítomnosťou modality, ktorá dosahovala vyššiu úspešnosť aj samostatne. Pomocou kombinácie viacerých modalít sa všeobecne zvýšila úspešnosť modelov. Najlepšou kombináciou je spojenie Ktrans a ADC, kedy sa AUC skóre zvýšilo o takmer 4% oproti modelom so samotnou modalitou. V druhom experimente sme pozorovali predikciu jedného nálezu z troch uhlov pohľadu, z dôvodu transformácie 2D vstupov do 3D informácie. Natrénovali sme tri modely, každý s inou orientáciou nálezu a predikcia konkrétneho nálezu bola kombinácia predikcií týchto modelov. Týmto spôsobom sa nám podarilo zvýšiť úspešnosť a dosiahnuť najlepšie AUC skóre 0.72. V treťom experimente sme zisťovali vplyv dvoch augmentačných techník na zlepšenie predikcie SCNN. Pomocou rigidnej deformácie na dáta sa zlepšila úspešnosť len zanedbateľne. Spojením rigidnej a nerigidnej deformácie náš model dosiahol o 2% lepšie výsledky oproti pôvodnému modelu.

Využitím architektúri Siamských sietí sme priniesli nový koncept do riešenia problematiky diagnostiky rakoviny prostaty. Naš model dosahuje sľubné výsledky, ktoré sú predpokladom pre ďalšie zdokonaľovanie tejto metódy. Táto práca bola taktiež prezentovaná ako výskumný článok na konferencií IIT SRC 2019.

V smere ďalšej práce by sme sa zamerali na lepšie odhadovanie zasiahnutého regiónu prostaty, kde sa nachádza nádor. Bolo by vhodné vykonať segmentáciu



nádoru a pozorovanie sústrediť len na túto zasiahnutú oblasť. V súčasnom riešení sa musí model vysporiadávať s nekonzistentnými regiónnymi. Ďalším smerom posunu je sústrediť sa na predspracovanie snímok a analyzovať už navrhnuté metódy normalizácie.

## Použitá literatura

---

- [1] ABRAHAM, B., AND NAIR, M. S. Computer-aided classification of prostate cancer grade groups from MRI images using texture features and stacked sparse autoencoder. *Computerized Medical Imaging and Graphics* 69 (2018), 60–68.
- [2] ATHANASIOU, L. S., FOTIADIS, D. I., AND MICHALIS, L. K. 9 - validation using histological and micro-ct data: Registration and inflation using ivus. In *Atherosclerotic Plaque Characterization Methods Based on Coronary Imaging*, L. S. Athanasiou, D. I. Fotiadis, and L. K. Michalis, Eds. Academic Press, Oxford, 2017, pp. 167 – 180.
- [3] BRAY, F., FERLAY, J., SOERJOMATARAM, I., SIEGEL, R. L., TORRE, L. A., AND JEMAL, A. Global cancer statistics 2018: Globocan estimates of incidence and mortality worldwide for 36 cancers in 185 countries. *CA: A Cancer Journal for Clinicians* 68, 6, 394–424.
- [4] BRON, D., GORE, M., AWADA, A., AND ROSEN, S. T. Editorial introductions. *Current Opinion in Oncology* 29, 5 (2017), v–vi.
- [5] CORNELISSE, D. An intuitive guide to convolutional neural networks. <https://medium.freecodecamp.org/an-intuitive-guide-to-convolutional-neural-networks-260c2de0a050>, Sep 14, 2016. Accessed: Dec 9, 2018.
- [6] DICKINSON, L., AHMED, H. U., ALLEN, C., BARENTSZ, J. O., CAREY, B., FUTTERER, J. J., HEIJMINK, S. W., HOSKIN, P. J., KIRKHAM, A., PADHANI, A. R., PERSAD, R., PUECH, P., PUNWANI, S., SOHAIB, A. S., TOMBAL, B., VILLERS, A., VAN DER MEULEN, J., AND EMBERTON, M. Magnetic resonance imaging for the detection, localisation, and characterisation of prostate cancer: Recommendations from a european consensus meeting. *European Urology* 59, 4 (2011), 477 – 494.
- [7] DOUKKALI, F. One-shot learning: Face recognition using siamese neural network. <https://towardsdatascience.com/>

one-shot-learning-face-recognition-using-siamese-neural-network-a13dcf739e, 2017. Accessed: Nov 29, 2018.

- [8] FARKAŠ, I., AND KRÁ, A. Úvod do teórie neurónových sietí.
- [9] FEHR, D., VEERARAGHAVAN, H., WIBMER, A., GONDO, T., MATSUMOTO, K., VARGAS, H. A., SALA, E., HRICAK, H., AND DEASY, J. O. Automatic classification of prostate cancer Gleason scores from multiparametric magnetic resonance images. *Proceedings of the National Academy of Sciences* 112, 46 (2015), E6265–E6273.
- [10] HE, K., ZHANG, X., REN, S., AND SUN, J. Deep residual learning for image recognition. *CoRR abs/1512.03385* (2015).
- [11] HIJAZI, S., KUMAR, R., AND ROWEN, C. Using Convolutional Neural Networks for Image Recognition. *Cadence Whitepaper* (2015), 1–12.
- [12] HIJAZI, S., KUMAR, R., AND ROWEN, C. Using Convolutional Neural Networks for Image Recognition. *Cadence Whitepaper* (2015), 1–12.
- [13] KOCH, G. Siamese Neural Networks for One-shot Image Recognition.
- [14] KOCH, G. Siamese Neural Networks for One-shot Image Recognition.
- [15] KYND, AND HIEDA, N. How neural networks are trained. [http://ml4a.github.io/ml4a/how\\_neural\\_networks\\_are\\_trained/](http://ml4a.github.io/ml4a/how_neural_networks_are_trained/), Sep 14, 2016. Accessed: Dec 9, 2018.
- [16] LE. Automated diagnosis of prostate cancer in multi-parametric MRI based on multimodal convolutional neural networks. *Phys. Med. Biol* (2018).
- [17] LEMATRE, G. Computer-Aided Diagnosis for Prostate Cancer using Multi-Parametric Magnetic Resonance Imaging.
- [18] LIU, S., ZHENG, H., FENG, Y., AND LI, W. Prostate cancer diagnosis using deep learning with 3D multiparametric MRI. 1013428.

- [19] MCNEAL, J., REDWINE, E., FREIHA, F., AND STAMEY, T. Zonal distribution of prostatic adenocarcinoma. correlation with histologic pattern and direction of spread. *The American journal of surgical pathology* 12 (01 1989), 897–906.
- [20] MEHRTASH, A., SEDGHI, A., GHAFOORIAN, M., TAGHIPOUR, M., CLARE, M., KAPUR, T., MOUSAVI, P., ABOLMAESUMI, P., FEDOROV, A., STATES, U., IMAGE, D., AND GROUP, A. Classification of Clinical Significance of MRI Prostate Findings Using 3D Convolutional Neural Networks.
- [21] RUDER, S. An overview of gradient descent optimization algorithms. <http://ruder.io/optimizing-gradient-descent/index.html#gradientdescentvariants>, 2016. Accessed: Nov, 19. 2018.
- [22] SARKAR, S., AND DAS, S. A review of imaging methods for prostate cancer detection: Supplementary issue: Image and video acquisition and processing for clinical applications. *Biomedical Engineering and Computational Biology* 7s1 (2016), BECB.S34255.
- [23] SCHRÖDER, F. H., HUGOSSON, J., ROOBOL, M. J., TAMMELA, T. L., CIATTO, S., NELEN, V., KWIATKOWSKI, M., LUJAN, M., LILJA, H., ZAPPA, M., DENIS, L. J., RECKER, F., BERENGUER, A., MÄÄTTÄNEN, L., BANGMA, C. H., AUS, G., VILLERS, A., REBILLARD, X., VAN DER KWAST, T., BLIJENBERG, B. G., MOSS, S. M., DE KONING, H. J., AND AUVINEN, A. Screening and prostate-cancer mortality in a randomized european study. *New England Journal of Medicine* 360, 13 (2009), 1320–1328. PMID: 19297566.
- [24] SIMONYAN, K., AND ZISSERMAN, A. Very deep convolutional networks for large-scale image recognition. *CoRR abs/1409.1556* (2014).
- [25] SOBECKI, P., ŻYCKA MALESA, D., MYKHALEVYCH, I., SKLINDA, K., AND PRZELASKOWSKI, A. Mri imaging texture features in prostate lesions classification.
- [26] SZEGEDY, C., LIU, W., JIA, Y., SERMANET, P., REED, S., ANGUELOV, D., ERHAN, D., VANHOUCKE, V., AND RABINOVICH, A. Going deeper with convolutions. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition 07-12-June-2015* (2015), 1–9.

- [27] VEEN, F. V. Neural network zoo. <http://www.asimovinstitute.org/neural-network-zoo/>, Apr 24, 2018. Accessed: Dec 5, 2018.
- [28] ŠIKUDOVÁ, E., ČERNEKOVÁ, Z., BENEŠOVÁ, W., HALADOVÁ, Z., AND KUČEROVÁ, J. *Počítačové videnie Detekcia a rozpoznávanie objektov*. Wikina, Praha, Praha, 2013.
- [29] WEINREB, J. C., BARENTSZ, J. O., CHOYKE, P. L., CORNUD, F., HAIDER, M. A., MACURA, K. J., MARGOLIS, D., SCHNALL, M. D., SHTERN, F., TEMPANY, C. M., THOENY, H. C., AND VERMA, S. Pi-rads prostate imaging – reporting and data system: 2015, version 2. *European Urology* 69, 1 (2016), 16 – 40.
- [30] WENG, Y.-A. C. W.-H. Learning Deep Representations of Medical Images using Siamese CNNs with Application to Content-Based Image Retrieval. 1–8.
- [31] YI, D., LEI, Z., AND LI, S. Deep metric learning for practical person re-identification. *Proceedings - International Conference on Pattern Recognition* (07 2014).

## A Plán práce na riešení projektu

---

### A.1 Prvá etapa projektu

- **1-2 týždeň** - Vyhľadanie datasetu pre diagnostiku rakoviny prostaty a jeho analyzovanie. (zobrazenie snímok, štatistiky z prídavných súborov k datasetu) Oboznámenie sa s oblasťou, oboznámenie sa s formátom DICOM a MRI technikami.
- **3-4 týždeň** - Oboznámenie sa s knižnicou ITK, na spracovanie snímok. (vyskúšať spracovať 3D MRI dáta z DICOM série). Štúdium príslušnej literatúry.
- **5-6 týždeň** - Predspracovanie 3D MRI dát pomocou ITK. (Extrahovanie jedného snímku s nálezom z 3D série, vyrezanie oblasti s léziou.) Vyhľadanie prác v oblasti diagnostiky RP založených na metódach hlbokého učenia.
- **7-8 týždeň** - vyskúšať augmentáciu dát (pomocou otáčania a zrkadlenia). Štúdium teórie neurónových sietí. Oboznámenie sa s knižnicami hlbokého učenia (Tensorflow, Keras).
- **9-10 týždeň** - spísanie analýzy neurónových sietí. Predspracovanie viacerých techník MRI z datasetu. Štúdium a implementovanie skúšobnej neurónovej siete pomocou knižnice Tensorflow.
- **11-12 týždeň** - Vytvorenie python scriptu pre spustenie C++ programu nad celým súborom dát. Opísanie súčasného stavu v našej oblasti. Spisovanie ďalších častí projektu.
- **13 týždeň** - Finalizácia analýzy a návrhu, odovzdanie práce.

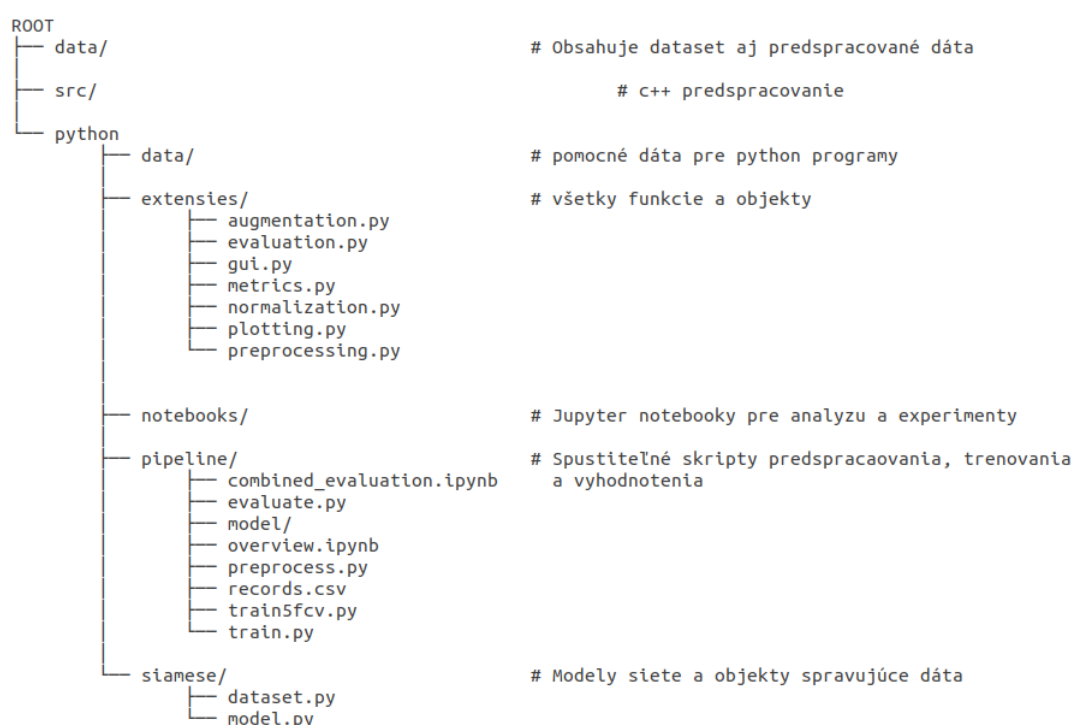
## A.2 Druhá etapa projektu

- **1-2 týždeň** - Implementovanie SCNN architektúry. Dokončovanie časti predspracovania, kombinácia modalít, testovanie a kontrola predspracovaných dát.
- **3-4 týždeň** - Spustenie prvého tréningu SCNN na našich dátach, základné modality. Implementácia pomocných funkcií, využívaných pri tréningu. Vyhodnotenie prvých modelov.
- **5-6 týždeň** - Pokračovanie tréningu nových modelov aj na kombinácií MRI modalít. Zdokonalenie tréningu a komplexnejšie vyhodnocovanie.
- **7-8 týždeň** - Začiatok písania článku IIT SRC. Vyhodnocovanie modelov. Predspracovať obrázky, aby sme dosiahli tri pohľady na nálezy. Tréning týchto modelov a ich vyhodnotenie.
- **9-10 týždeň** - Zdokumentovať etapu návrhu a implementácie.
- **11-12 týždeň** - Zdokumentovať etapu testovanie, dokončenie spisovanie práce. Upravenie dokumentu do finálnej podoby a odovzdanie práce.

## B Technická dokumentácia

### B.1 Používateľská príručka

V tejto časti si vysvetlíme základné požiadavky na použitie nášho nástroja a jeho štruktúru. Náš nástroj pozostáva z viacerých adresárov, python súborov a Jupyter notebookov. Na obrázku 20 môžete vidieť štruktúru základných súborov potrebných na prácu s nástrojom. Súčasťou riešenia je aj mnoho Jupyter notebookov, ktoré sú samovysvetľujúce.



Obr. 20: Adresár základných súborov nášho projektu.

Pre opätovné spustenie je potrebné si do svojho prostredia nainštalovať jazyk Python, aj so sadou jeho knižníc. Potrebné python knižnice: Numpy, Pandas, SimpleITK, Tensorflow, augmentation, imgaug, matplotlib a scikit-learn.

Do adresára *data/* je potrebné stiahnuť dataset. Adresár slúži na uchovávanie nespracovaných dát, ale taktiež sa tu ukladajú aj predspracované dáta. Dataset PROSTA-



TEx je možné stiahnuť zo stránky [https://wiki.cancerimagingarchive.net/display/Public/SPIE-AAPM-NCI+ PROSTATEx+Challenges](https://wiki.cancerimagingarchive.net/display/Public/SPIE-AAPM-NCI+PROSTATEx+Challenges).

Keď sú už naše dáta na správnom mieste, môžeme spúšťať python skripty, ktoré využívajú mnoho ďalších funkcií a objektov. Pre spustenie riešenia je dôležitý adresár *python/pipeline/*, kde sa nachádzajú všetky spustiteľné súbory. Popíšeme si po krokoch, čo je potrebné k spusteniu jednotlivých častí riešenia.

### B.1.1 Predspracovanie dát

Pri predspracovaní našich dát je potrebné spustiť súbor *python/pipeline/preprocess.py*, ktorý obsahuje python skript. Najskôr je potrebné tento súbor otvoriť a definovať parametre úpravy dát. Na obrázku je vidieť premenné *root\_path*, ktorá obsahuje adresár, kde máme uložený dataset. Ak sme nenašli pôvodné miesto uloženia, táto premenná môže ostať nezmenená. Podobne aj *path\_to\_data* je cesta k adresáru, kam sa uložia upravené dáta. Následne k premennej *modality* priradíme jednu z hodnôt *t2tsetra*, *adc*, *ktrans*, *cbval*, čo znamenajú nasledovné modality T2W, ADC, Ktrans a Bval. Zvyšné premenné *new\_spacing*, *orientation*, *patch\_size*, ktoré hovoria o rozlíšení výsledného obrázku v mm, ďalej z akého pohľadu chceme obrázok nadobudnúť, tam môže byť *t,s,a*. V poslednej premennej definujeme veľkosť výsledného obrázku, ktorá sa udáva v pixeloch. Ak tretie číslo vo vektore je 1, jedná sa o 2D obrázok, ak je viac ako 1, získame 3D obrázok.

```
# Define parameters of preprocessing
root_path = '../..data/PROSTATEx'
path_to_data = '../..data/'
modality = 'ktrans'
new_spacing = (0.5,0.5,0.5)
orientation = 't'
patch_size = (40,40,1)
```

Obr. 21: Definovanie parametrov predspracovania dát.

Výsledné dáta podľa príkladu v obrázku 21 budú uložené do adresára s menom *../..data/ktrans/t/40x40x1/* a pri nich bude pomocný súbor s informáciami o nich s názvom *info.csv*. Skript môžeme v tomto momente spustiť z príkazového riadku.

```

# iterate through all findings
index = 0
for idx, (_, row) in enumerate(findings.iterrows()):
    print(row['ProxID'])

    # Load image by modality
    if modality == 'ktrans':
        path_to_image = my.get_ktrans_path(row['ProxID'], path_to_data)
        image = sitk.ReadImage(path_to_image)
    else:
        #
        path_to_image = my.get_path(row['ProxID'], modality, root_path)
        fixed_series_filenames = sitk.ImageSeriesReader_GetGDCMSeriesFileNames(path_to_image)
        image = sitk.ReadImage(fixed_series_filenames)

    # fit normalizer
    normalizer = normalization.ScaleNormalization()
    normalizer.fit(sitk.GetArrayFromImage(image))
    # resample image to same spacing in all directions
    image = my.resample_image_to_spacing(image, new_spacing, sitk.sitkLinear)

    # get center of lesion from dataset
    center = [float(x) for x in row['pos'].split()]
    # extract region of interest
    for i, offset in enumerate(offsets):
        volume = my.get_patch_from_image(image, patch_size, center, orientation)
        # normalise data
        print(volume.dtype)
        volume = normalizer.normalise(volume)

    # define informations about image
    file_name = row['ProxID'] + '_' + str(index) + '_' + row['zone'] + '.nii'
    new_df.loc[index] = row[['ProxID', 'fid', 'zone', 'ClinSig']]
    new_df.loc[index, 'name'] = file_name
    new_df.loc[index, 'ID'] = idx
    new_df.loc[index, 'normalization'] = normalizer.name
    # save image to file
    my.save_image(volume, os.path.join(target_path, file_name))

    index += 1

# save informations about images to the directory
new_df.to_csv(os.path.join(target_path, 'info.csv'))

```

Obr. 22: Hlavný cyklus spracovania snímku s nálezom rakoviny.

Na obrázku 22 môžete vidieť iterovanie cez informácie o všetkých nálezoch v premennej `findings`. Najprv je pre konkrétny záznam nájdená cesta k súboru so snímkom pomocou funkcie `my.get_path(row['ProxID'], modality, root_path)`. Následne je prečítaný obrázok z tejto cesty pomocou funkcie `sitk.ReadImage()` z knižnice `SimpleITK`.

Normalizácia je aplikovaná pomocou objektu `ScaleNormalization()`, ktorý je aj s ďalšími normalizačnými technikami definovaný v súbore `python/extended/normalization.py`, ktorý je možné nastaviť pomocou metódy `fit()` a neskôr aplikovať pomocou metódy `normalise()`.

Interpolovanie na rovnaké rozlíšenie prebieha vo funkcii `my.resample_image_to_spacing()`. Na obrázku 23 môžete vidieť túto funkciu, ktorá vráti už upravený obrázok.

```
def resample_image_to_spacing(image, spacing, interpolator, report = False):
    #compute new size of image

    phys_size = [ org_spacing*(org_size - 1) for org_spacing,org_size in zip(image.GetSpacing(),image.GetSize())]
    new_size = [ int(round(phys / space)) + 1 for phys,space in zip(phys_size, spacing)]

    #Set up resample
    resample = sitk.ResampleImageFilter()
    resample.SetInterpolator(interpolator)
    resample.SetOutputDirection(image.GetDirection())
    resample.SetOutputOrigin(image.GetOrigin())
    resample.SetOutputSpacing(spacing)
    resample.SetSize(new_size)

    newimage = resample.Execute(image)
    if report:
        print('Size:{} to {}, Spacing: {} to {}'.format(image.GetSize(),
                                                         newimage.GetSize(),image.GetSpacing(),newimage.GetSpacing()))

    return newimage
```

Obr. 23: Funkcia interpolovania snímku na jednotné rozlíšenie.

Keď už máme dointerpolovaný obrázok, extrahujeme z neho oblasť záujmu, kde sa nachádza pozorovaná lézia. Extrakciu zabezpečuje funkcia: `get_patch_from_image()`, ktorá je zobrazená v obrázku 24.

```

def get_patch_from_image(image = None, size = None, center = None, orientation = 't', offset = [0,0,0]):
    # change shape by orientation
    center = np.add(center, offset)
    if orientation == 's':
        size = (size[0],size[2],size[1])
    elif orientation == 'a':
        size = (size[2],size[1],size[0])
    elif orientation == 't':
        pass
    else:
        raise Exception('Wrong orientation type! Chose from this tree: t,s,a')
    # transform physical coordinates to pixel coordinates
    center_idx = np.array(image.TransformPhysicalPointToContinuousIndex(center)) + 0.5
    vol = sitk.GetArrayFromImage(image)
    image_size = image.GetSize()
    # count boundaries of new image, if boundaries are out of then Exception
    max_axis = (int(round(idx + lenght / 2)) for idx, lenght in zip(center_idx, size))
    min_axis = (int(round(idx - lenght / 2)) for idx, lenght in zip(center_idx, size))
    boundaries = list(zip(max_axis, min_axis))
    thresholds = list(treshold + 1 >= ma and 0 <= mi for treshold, (ma, mi) in zip(image_size, boundaries))
    if False in thresholds:
        raise Exception('Boundaries out of image!')
    # reverse boundaries because vol has reversed axes
    boundaries = list(reversed(boundaries))
    # get pixels in boundaries
    patch = vol[boundaries[0][1]:boundaries[0][0], boundaries[1][1]:boundaries[1][0], boundaries[2][1]:boundaries[2][0]]
    # transform arrays for different orientations to shape for image with shape X/Y/Z
    if orientation == 's':
        patch = np.flip(patch, 1)
        patch = np.swapaxes(patch, 0, 1)
    elif orientation == 'a':
        patch = np.swapaxes(patch, 0, 1)
        patch = np.swapaxes(patch, 0, 2)
        patch = np.flip(np.flip(patch, 2), 0)
    # when we took 2d image then transform dimension
    if 1 in size:
        patch = patch[0]
    return patch

```

Obr. 24: Funkcia interpolovania snímku na jednotné rozlíšenie.

Výstupom z tejto funkcie je finálny obrázok, nad ktorým ešte aplikujeme normalizáciu a uložíme do súboru v adresári, ktorý sme spomínali vyššie. Vytvorí sa aj záznam pre tento obrázok, ktorý sa pridá k zvyšným.

### B.1.2 Trénovanie modelu

Pre začatie tréovania modelu je potrebné spustiť súbor *python/pipeline/preprocess.py*, ktorý obsahuje python skript. Najskôr je potrebné tento súbor otvoriť v textovom editore a definovať niektoré parametre. Na obrázku 25 je vidieť premenné `batch_size`, `train_iter`, `step`, ktoré definujú veľkosť tréovacej dávky, počet iterácií a počet iterácií, po ktorých sa bude model vizualizovať. Zoznam `modalities` obsahuje cesty k obrázkom, ktoré chceme použiť pri tréovaní. Boolean premenná `augmentation` značí, či chceme pri tréovaní použiť augmentáciu.

```
flags = tf.app.flags
FLAGS = flags.FLAGS
flags.DEFINE_integer('batch_size', 256, 'Batch size.')
flags.DEFINE_integer('train_iter', 1400, 'Total training iter')
flags.DEFINE_integer('step', 50, 'Save after ... iteration')
flags.DEFINE_string('path_to_data', '../data/', 'Path to data')
modalities= ['adc/t/20x20x1', 'cbval/t/20x20x1', 'ktrans/t/20x20x1']
augmentation = True
```

Obr. 25: Základné premenné pre proces tréovania modelu.

Na obrázku 26 môžeme vidieť načítanie dát pomocou objektu `DataLoader`, ktorý je definovaný v súbore *python/siamese/dataset.py*. Tento objekt sa stará o načítanie dát a vytvorenie generátoru pre 5-násobnú krížovú validáciu. V prípade ak používame viac ako jednu modalitu, `DataLoader` vytvára vstup kombináciou týchto modalít do jedného viackanálového vstupu.

```
# load data and combine them if more than 1 modality
if len(modalities) > 1:
    loader = DataLoader(FLAGS.path_to_data, modalities)
    loader.load_data()
    loader.combine_channels(modalities)
    kfold = loader.k_fold(subdir = 'combined', num = 5)
    data_shape = loader.get_shape('combined')
else:
    loader = DataLoader(FLAGS.path_to_data, modalities)
    loader.load_data()
    kfold = loader.k_fold(subdir = modalities[0], num = 5)
    data_shape = loader.get_shape(modalities[0])
```

Obr. 26: Načítavanie tréovacích dát.

Pomocou vytvoreného `kfold` generátora iterujeme cez jednotlivé rozdelenia dát na tréningovú a testovaciu sadu nasledovne: `for X_train, X_test, y_train, y_test in kfold:`. Pomocou týchto dát inicializujeme objekt `Dataset`, ktorý má metódu `get_siamese_batch` pomocou ktorej generujeme dávky pre našu sieť. Ak využívame augmentáciu inicializujeme objekt `augmentor`.

```
dataset = Dataset()
dataset.images_train = X_train
dataset.images_test = X_test
dataset.labels_train = y_train
dataset.labels_test = y_test
## AUGMENTATION
if augmentation:
    augmentor = aug.Elastic2Augmentor(X_train)
    row.loc[0, 'augmentation'] = augmentor.name
else:
    augmentor = False
    row.loc[0, 'augmentation'] = False

next_batch = dataset.get_siamese_batch
```

Obr. 27: Definovanie objektu `Dataset`, `Augmentor`.

Definovanie modelu SCNN prebieha nasledovne ako na obrázku 28. Vytvoríme vstupy, zavoláme funkciu `xmas_model` ktorá definuje náš CNN model. Tieto modely prepojíme stratovou funkciou `contrastive_loss`.

```
# Setup network
model = xmas_model
model_name = "xmas"
placeholder_shape = [None] + list(data_shape[1:])

left = tf.placeholder(tf.float32, placeholder_shape, name='left')
right = tf.placeholder(tf.float32, placeholder_shape, name='right')
with tf.name_scope("similarity"):
    label = tf.placeholder(tf.int32, [None, 1], name='label') # 1 if same, 0 if different
    label_float = tf.to_float(label)
margin = 0.5
left_output = model(left, reuse=False)
right_output = model(right, reuse=True)
loss = contrastive_loss(left_output, right_output, label_float, margin)
# -- Setup network --
```

Obr. 28: Špecifikácia modelu SCNN a jeho optimalizácia.

Na obrázku 29 môžete vidieť definované `xmas_model` a `contrastive_loss` funkcie, ktoré boli použité pri vytvorení SCNN modelu.

```
def xmas_model(input, reuse=False):
    """
    CNN model inspired by: Liu, Saifeng et al. "Prostate cancer diagnosis using deep learning with 3D multiparametric MRI."
    """
    with tf.name_scope("model"):
        with tf.variable_scope("conv1") as scope:
            net = tf.contrib.layers.conv2d(input, 32, [3, 3], activation_fn=tf.nn.relu, padding='SAME',
                                           weights_initializer=tf.contrib.layers.xavier_initializer_conv2d(), scope=scope, reuse=reuse)

        with tf.variable_scope("conv2") as scope:
            net = tf.contrib.layers.conv2d(net, 32, [3, 3], activation_fn=tf.nn.relu, padding='SAME',
                                           weights_initializer=tf.contrib.layers.xavier_initializer_conv2d(), scope=scope, reuse=reuse)
            net = tf.contrib.layers.max_pool2d(net, [2, 2], padding='SAME')

        with tf.variable_scope("conv3") as scope:
            net = tf.contrib.layers.conv2d(net, 64, [3, 3], activation_fn=tf.nn.relu, padding='SAME',
                                           weights_initializer=tf.contrib.layers.xavier_initializer_conv2d(), scope=scope, reuse=reuse)

        with tf.variable_scope("conv4") as scope:
            net = tf.contrib.layers.conv2d(net, 64, [3, 3], activation_fn=tf.nn.relu, padding='SAME',
                                           weights_initializer=tf.contrib.layers.xavier_initializer_conv2d(), scope=scope, reuse=reuse)
            net = tf.contrib.layers.max_pool2d(net, [2, 2], padding='SAME')

        with tf.variable_scope("fc1") as scope:
            net = tf.contrib.layers.flatten(net)
            net = tf.contrib.layers.fully_connected(net, 256, activation_fn=tf.nn.relu, scope=scope, reuse=reuse)

        with tf.variable_scope("fc2") as scope:
            net = tf.contrib.layers.fully_connected(net, 8, activation_fn=tf.nn.relu, scope=scope, reuse=reuse)

    return net

def contrastive_loss(model1, model2, y, margin):
    with tf.name_scope("contrastive-loss"):
        distance = tf.sqrt(tf.reduce_sum(tf.pow(model1 - model2, 2), 1, keepdims=True))
        similarity = y * tf.square(distance) # keep the similar label (1) close to each other
        # give penalty to dissimilar label if the distance is bigger than margin
        dissimilarity = (1 - y) * tf.square(tf.maximum((margin - distance), 0))
        return tf.reduce_mean(dissimilarity + similarity) / 2
```

Obr. 29: Definícia modelu CNN a stratovej funkcie.

Keď je náš model zostavený a parametre sú nastavené, začína proces optimalizácie modelu, ako môžete vidieť na obrázku 31. Optimalizátor je vytvorený ako `tf.train.AdamOptimizer(0.00001)`, kedy môžeme upraviť rýchlosť učenia. Počas procesu tréningu sa vykresľujú dva grafy. Prvý ukazuje klesanie stratovej funkcie a druhý ukazuje rast AUC skóre. Po skončení optimalizácie je model uložený do adresára `pyhon/pipeline/model` pre neskoršie využitie. Model je automaticky po dotrenovaní vyhodnotený a aj s výsledkami je jeho záznam uložený do súboru `pyhon/pipeline/records.csv`.

```

global_step = tf.Variable(0, trainable=False)
train_step = tf.train.AdamOptimizer(0.00001).minimize(loss, global_step=global_step)

# Start Training
saver = tf.train.Saver()
with tf.Session() as sess:
    sess.run(tf.global_variables_initializer())

    #setup tensorboard
    tf.summary.scalar('step', global_step)
    tf.summary.scalar('loss', loss)
    for var in tf.trainable_variables():
        tf.summary.histogram(var.op.name, var)
    merged = tf.summary.merge_all()
    writer = tf.summary.FileWriter('train.log', sess.graph)

    #train iter
    for i in range(FLAGS.train_iter):
        batch_left, batch_right, batch_similarity = next_batch(FLAGS.batch_size, augmentor = augmentor)

        _, l_train, summary_str = sess.run([train_step, loss, merged],
                                           feed_dict={left:batch_left, right:batch_right, label: batch_

        batch_left, batch_right, batch_similarity = next_batch(FLAGS.batch_size, source = 'test')
        l_test= sess.run(loss,
                        feed_dict={left:batch_left, right:batch_right, label: batch_

        batch_train_losses.append(l_train)
        batch_test_losses.append(l_test)

        writer.add_summary(summary_str, i)
        print("\r#%d - Loss"%i, l_train, "Test_Loss ", l_test)

    if (i + 1) % FLAGS.step == 0:
        #generate test
        # TODO: create a test file and run per batch

        a = dataset.test_oneshot(sess, left_output, left, 4, 100)
        b = dataset.test_oneshot(sess, left_output, left, 4, 100, 'train')
        print('Accuracy train: ', b)
        print('Accuracy test: ', a)
        accuracy_test.append(a)
        accuracy_train.append(b)
        loss_test.append(np.sum(batch_test_losses) / len(batch_test_losses))
        loss_train.append(np.sum(batch_train_losses) / len(batch_train_losses))

```

Obr. 30: *Proces optimalizácie modelu SCNN.*

### B.1.3 Vyhodnotenie modelu

Kontrola, poprípade ďalšie experimenty s modelom alebo kombinácia predikcií viacerých modelov (3 orientácie nálezu) prebieha pomocou Jupyter notebooku `pyhon/pipeline/combined_evaluation.ipynb`. V tomto notebooku nájdete výpis všetkých doposiaľ natrénovaných modelov.



```

# prediction of the most similar
def siamese_predict(train_feat, search_feat, dataset, n = 10, distance = 'cosine'):
    dist = cdist(train_feat, [search_feat], distance)
    rank = np.argsort(dist.ravel())

    labels = dataset.labels_train[rank[:n]]
    c = Counter(labels)
    dictionary = dict([(i, c[i] / len(labels) * 100.0) for i in c])
    a = max(dictionary.items(), key=operator.itemgetter(1))[0]
    b = labels[0]
    return a

# prediction by percentage of positives in top similar
def threshold_predict(train_feat, search_feat, dataset, threshold = 0.5, n = 10, distance = 'cosine'):

    dist = cdist(train_feat, [search_feat], distance)
    rank = np.argsort(dist.ravel())

    labels = dataset.labels_train[rank[:n]]
    counter = Counter(labels)
    percentage_dict = dict([(i, counter[i] / len(labels)) for i in (0,1)])

    if percentage_dict[1] >= threshold:
        return 1
    else:
        return 0

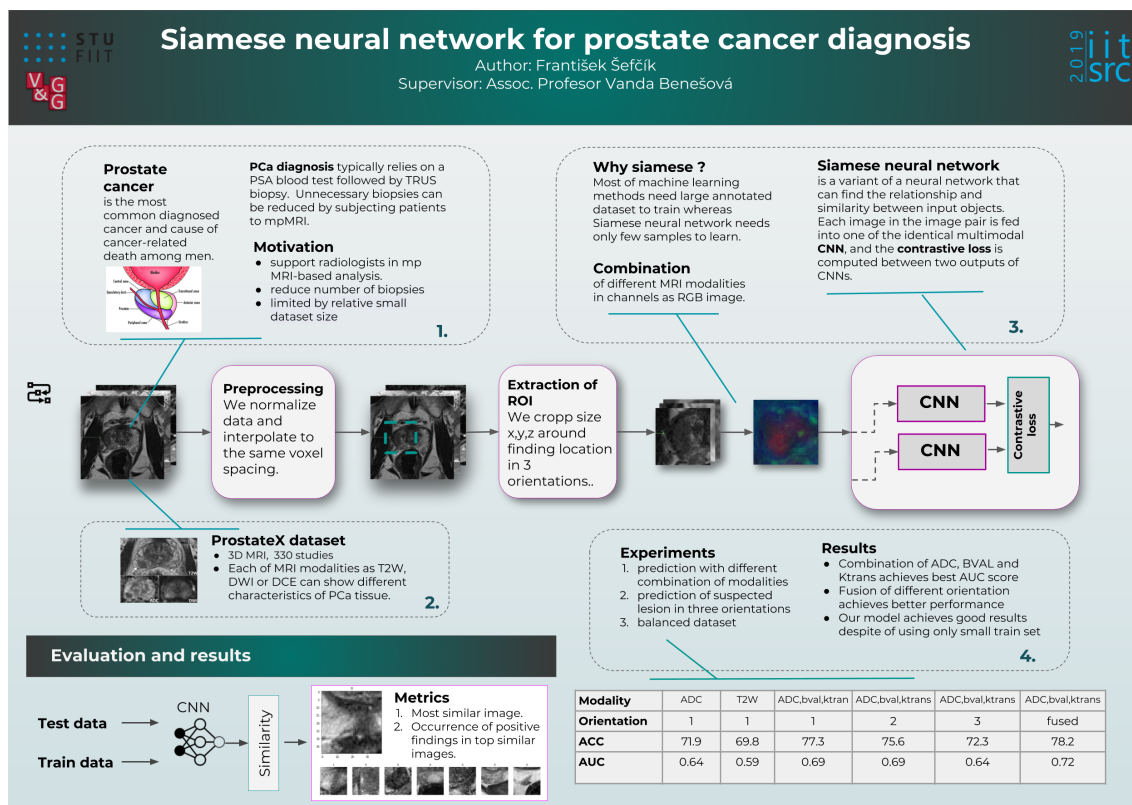
def get_separated(y, y_pred):
    dictionary = dict({})
    # ma byt pozitivny a aj je predikovaný ako pozitivny
    dictionary['tp'] = np.where([x==y and x == 1 for (x,y) in zip(y,y_pred)])[0]
    # ma byt pozitivny je predikovaný ako negativny
    dictionary['fn'] = np.where([x!=y and x == 1 for (x,y) in zip(y,y_pred)])[0]
    # ma byt negativny a aj je predikovaný ako negativny
    dictionary['tn'] = np.where([x==y and x == 0 for (x,y) in zip(y,y_pred)])[0]
    # ma byt negativny je predikovaný ako pozitivny
    dictionary['fp'] = np.where([x!=y and x == 0 for (x,y) in zip(y,y_pred)])[0]
    return dictionary

# function to find most similar images to test one
def show_most_similar(train_feat, search_feat, dataset, idx = 0, n = 10):
    dist = cdist(train_feat, [search_feat[idx]], 'euclidean')
    rank = np.argsort(dist.ravel())
    labels = dataset.labels_train[rank[:n]]
    plt.title(str(dataset.labels_train[idx]))
    plt.imshow(dataset.images_test[idx, :, :], cmap = 'gray')
    show_image(rank[:n], dataset.images_train, dataset.labels_train)

```

Obr. 31: Prvé dve funkcie určujú triedu pre testovací obrázok. Zvyšné dve funkcie pomáhajú vizuálne kontrolovať predikciu modelov.

## C Článok na IIT SRC



Obr. 32: Plagát na konferencii IIT SRC 2019.

# Processing of visual medical data by computer vision methods and deep neural networks

František ŠEFČÍK\*

*Slovak University of Technology in Bratislava  
Faculty of Informatics and Information Technologies  
Ilkovičova 2, 842 16 Bratislava, Slovakia  
frantisek.sefcik@gmail.com*

**Abstract.** Prostate cancer is the most common diagnosed cancer and cause of cancer-related death among men. Computer Aided Diagnosis (CAD) systems are used to support radiologists in multiparametric Magnetic Resonance (mpMR) image-based analysis, in order to distinguish clinically significant prostate cancer lesions from benign ones. The methods of deep learning and artificial neural networks, have great results in image processing, and are increasingly coming to the forefront. Most of related studies investigated CAD systems for PCa diagnosis based on single supervised convolution neural network (CNN), which usually relies heavily on a large scale annotated dataset for training. A common problem of these methods is lack of accessible annotated data which limit their approach. To learn image representation with less supervision, we propose a deep Siamese CNN (SCNN) architecture that can be trained with only binary image pair information.

## 1 Introduction

Prostate cancer is the second most common type of cancer and fifth leading cause of death in men. It is estimated that there was almost 1.3 million new cases of prostate cancer and 359,000 associated deaths worldwide in 2018 [2]. The clinical method of early detection of PCa relies on Prostate Specific Antigen (PSA) blood test followed by transrectal ultrasound (TRUS) biopsy. The diagnosis by PSA and TRUS biopsy reduce PCa mortality by 20-30% but it has a relatively low specificity that leads to a repeat number of biopsies that affects the success of the patient's treatment and comfort [8]. Another major risk with biopsy is the diagnosis of clinically insignificant or indolent cancer. Unnecessary biopsies can be reduced by subjecting patients to mpMRI before performing ini-

tial biopsy. Multiparametric MRI can also minimize over-diagnosis of indolent PCa and improve diagnosis of clinically significant PCa however the manual reading of mpMRI by radiologist are time consuming due to the number of images and also largely dependent on the experience of the radiologist.

A number of works based on automatic analysis of mpMRI has been invented to help radiologist distinguish between clinically significant and indolent PCa. In recent years, deep convolution neural networks (CNN) as one of deep learning techniques are widely used for various computer vision task when a sufficient number of annotated data is provided. However effectiveness of CNNs for medical imaging is limited by relative small sample size and quiet different image characteristics from natural images. Augmenting limited training data with a sufficient number and variety of data is critical step to train model. Incorrect augmentation can lead to unsuccessful result. Another way avoiding data augmentation may by neural network architecture that does not need a robust dataset to train.

In this paper we present a siamese neural network (SCNN) developed for the PROSTATEx challenge, prepared for task of classification of clinically significant PCa findings in mpMRI. Our deep SCNN learns to differentiate an image pair by evaluating the similarity and relationship between the given images. Each image in the image pair is fed into one of the identical multimodal CNN, and the contrastive loss is computed between two outputs of CNNs. We compare our proposed SCNN method with single CNNs presented in several studies that use massive augmentation techniques for training.

---

\* Bachelor study programme in field: Informatics

Supervisor: Assoc. Professor Vanda Benešová, Institute of Computer Engineering and Applied Informatics, Faculty of Informatics and Information Technologies STU in Bratislava

## 2 Related work

Research into prostate cancer diagnosis has brought many promising solutions. Generally speaking, there are methods of formulating a diagnosis of PCa as a series of classification tasks based on supervised learning. Existing methods differ from each other in terms of features and classifiers used for representing and differentiating images of prostate tissues, different combinations of modalities and methods for multimodal fusion.

Multiparametric MRI has become one of most used choices for non-invasive and more accurate way for PCa diagnosis. Lemaitre (2016) [5] described a comprehensive survey of CAD systems for the diagnosis of PCa on mpMRI. Prostate Imaging Reporting and Data System Version 2.0 (PI-RADS 2.0) defines that mpMRI for PCa detection and diagnosis typically include T2-weighted imaging (T2W), diffusion-weighted imaging (DWI) and dynamic contrast-enhanced (DCE) MRI. Various combinations of MRI modalities have been investigated. For instance, Le et al. (2017) [4] combined the T2W and apparent diffusion coefficient (ADC) from DWI, Liu et al. (2017) [6] combined DWI, ADC, T2W and K trans from DCE-MRI to different type of inputs as the RGB channels and Mehrtash et. al. (2017) [7] combined besides K trans and ADC, maximum b-value from DWI.

For simplification and easier calculation of feature map is needed to extract region of interest (ROI) with PCa lesion from image volume. Many works exist where extraction is done manually by the radiologist as in [4], then several studies obtained ROI by segmentation of PCa lesion. In [6] the lesion region was obtained through region growing and morphological operations, using DWI. One of the solutions was presented in [1, 7], where ROI of size  $x/y/z$  is cropped around finding location and all lesions were approximately fit in this area.

However, being able to augment limited training data with a sufficient number and variety of data for medical tasks is a critical factor, which was investigated in [4, 5]. Since the prostate and PCa are non-rigid objects, conventional data augmentation methods as rotation, translation, scaling and flipping, and applying global rigid transformation to the original data, might not be sufficient to increase the variety of data for CNN training. [4, 5]. Including data augmented by non-rigid deformation could simulate more reasonable example lesions of different shapes, increasing both the number and variety of data. In study [4] demonstrated that from rigid and non-rigid, combination of both achieve the best results. Partial augmentation could be provided through spatial rotation around center of lesion and slicing. [6]

Classification and feature selection is the most important step for quality prediction. Most of the existing techniques use hand-crafted features for diagnosing PCa [5]. In recent years, many studies have explored CNNs as generic feature extractor for PCa diagnosis. Le et. al. [4] compared three popular state-of-the-art CNNs architectures with different depth but experimental results show that the performance difference among the three architectures is relatively small. Details of existing works with CNN that distinguish clinically significant PCa from indolent ones are consolidated in Table 1.

*Table 1. Review of PCa diagnosis methods based on CNN*

Author	Modalities	Augmentation	AUC
Liu [6]	T2W,DWI, ADC,Ktrans	3D rotation, slicing	0,89
Le [4]	T2W,ADC	rigid, non-rigid	0,909
Mehrt. [7]	ADC,BVAL, Ktrans	flipping, translations	0,85

## 3 Materials and methods

### 3.1 Dataset

The dataset was collected by the Cancer Imaging Archive (TCIA) and it was used as training data for PROSTATEx challenge. It consists of 204 subjects with 330 suspicious lesion findings. From 330 lesions is 254 indolent and 76 clinically significant. For each subject, there were Diffusion Weighted Images (DWI), Apparent Diffusion Coefficient (ADC) maps, Ktrans, as well as T2 Weighted Images (T2WI). The dataset contains complementary information for each lesion as the location of finding, label (cancerous, non-cancerous) and prostate anatomy region where the lesion is located.

### 3.2 Methods

The architecture of a proposed method for automatic PCa diagnosis is shown in the Figure 1. The method includes preprocessing, where T2W and ADC sequences are registered to reduce patients motion during data acquisition and normalize intensities to reduce intensity variations among all patients. Then we extract region of interest (ROI) from image sequences in tree different orientation. The augmentation of data is provided for one of our experiments. For classification, we propose SCNN architecture which consists of two identical CNNs. Final prediction is done through voting of tree SCNN, each trained for one of the tree orientation (coronal, sagittal, traversal). Multimodal SCNN takes pair of registered and normalized T2W-ADC as input. In the following, we present details of each module.

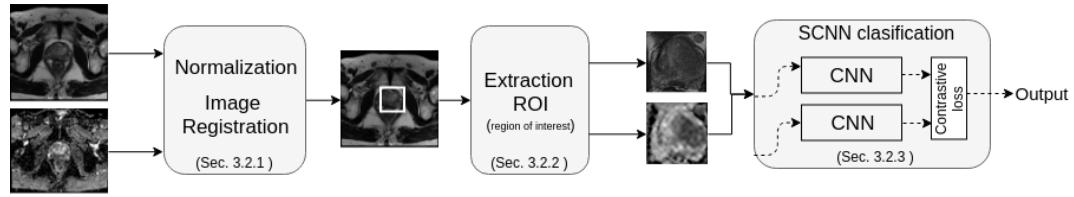


Figure 1. Architecture of automated system for PCa diagnosis.

### 3.2.1 Registration and normalization

Image registration methods have become an important role in most of systems that work with more than one MRI modality. Each of MRI modalities as T2W, DWI or DCE can show different characteristics of PCa and fusion of more than one modality lead to a better diagnosis of PCa. Before we start classification and feature extraction on mpMRI is necessary to eliminate the differences between sequences of mpMRI caused by patients movement and distortion during acquisition of MRI. To minimize this misalignment errors we applied rigid registration where target ADC data are aligned corresponding to fixed T2W image.

Other common problems with using mpMRI are inhomogenous intensities for each modality among patients. This inhomogeneity is caused by different MRI protocols and parameter settings during data acquisition. To correct intensities we first applied intensity normalization to rescale intensities of both ADC and DWI to [0-1].

### 3.2.2 Extracting ROI

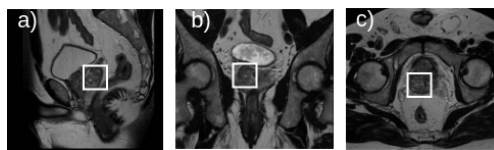


Figure 2. Three different orientations: a) sagittal, b) coronal, c) axial in T2W MRI.

For successful classification and feature extraction is necessary to focus observation only on the region where the finding is located. In many studies, they segment prostate gland where a suspected lesion is located. The second approach the region surrounding lesion is extracted what we consider as the best approach to our problem. Segmentation of lesion is provided in several works to obtain an accurate region of prostate cancer. In our study, we propose extract region surrounding the center of the lesion based on information of finding location provided in the dataset. From each of T2W and ADC sequences, we extract

tree 2D slices with same square size. Each of the slices is obtained under tree different angles that was done through 3D rotation as is shown in Figure 2. This multi-view techniques reduce the 3D problem of PCa to 2D problem and enables the insertion of 3D information in the 2D inputs.

### 3.2.3 Model and training

For classification, we propose a siamese neural network that has been successfully used for one-shot image classification [3]. Application for medical imaging was covered in the study [9] where experimental results show that deep SCNN is comparable to the state-of-the-art single supervised CNN, and requires much less supervision for training. We construct deep SCNN for the task to distinguish between indolent and clinically significant PCa lesion.

Deep SCNN architecture is a variant of a neural network that can find the relationship and similarity between the input objects. Our architecture has two symmetric CNN subnetworks sharing the same parameters and weights and during training, they are updating equally. Each subnetwork takes the ADC-T2W pair as two channels input and we get feature vector as the output. The subnetwork is build with 5 layers of convolutional layers and max pooling followed. Two last convolutional layers has function of fully connected layer because convolutional operation is faster on GPU. The sizes of convolution kernels are following [7x7],[5x5],[3x3],[1x1],[1x1].

For training the network, we use the stochastic gradient descent algorithm with the Adam update rule, a mini-batch size of 256, and a contrastive loss function. We initialized the CNN weights randomly using the Xavier method. Contrastive loss function calculates the similarity between the two images. We randomly take a pair of two ADC-T2W images, each contains one suspicious lesion and we label this pair either it is from the same class or not. Each image in the image pair is fed into one of the identical CNNs, and the contrastive loss is computed between two outputs of CNNs.

With above-mentioned method, we will train three SCNN models each for different lesion orientation. Then the final prediction for each lesion is calculated

as predictions of those SCNN made over different orientations.

### 3.2.4 Augmentation

In this study, we aim to fully explore the capability of deep SCNN trained only with data available in the dataset. For the reason that the dataset is imbalanced (254 indolent and 76 clinically significant) we examine the impact of augmentation for the reason to balance dataset. As the study [4] demonstrated that applying both rigid and non-rigid deformation methods can achieve the best performance. We decided to apply the rigid transformation, including random translation, rotation and scaling, flipping and non-rigid deformation for data augmentation.

## 4 Experimental results

Whereas the output of the trained model is not final prediction we evaluate model by comparison of the test image with a set of known images. As first the model counts feature vectors for all train and test images. Individually for each of the test images, we get Euclidean distances with train images. We proposed two approaches to make a prediction over this distances when we label test image as the most similar image (image with the smallest distance to reference image) and the second one if among 10 most similar images are more than 40% of positive findings then label it as positive. The second approach has got more accurate results so we use it as our reference method.

In Table 2 we report the results of our experiments. We applied preprocessing on the three of MRI modality where outputs are normalized images of size 40x40 pixels with identical voxel spacing. We evaluate the accuracy of our model with single MRI modality and a combination of them in channels. As the Table 2 shows, the best accuracy achieved single ADC. We expected that the combination of MRI modalities can describe more PCa characteristics and help for prediction. This approach achieved similar accuracy with single modality but we believe in improvement after refining and tuning the model. In the next experiment, we applied rigid and non-rigid augmentation to balance dataset. Both helped to minimize overtraining of the models but did not improve accuracy compared to no augmentation models.

Table 2. Review of experimental results

Modality	Accuracy	AUC
ADC	78.3	68.8
T2W	71.1	66.8
Ktrans	73.1	65.4
T2W + ADC	72.4	61.7
T2W + ADC + Ktrans	75.2	66.7

## 5 Conclusion and future work

In this paper we proposed a deep SCNN method for task of automatic classification of prostate cancer lesions, using the 3D mpMRI. Our method uses advantages of deep CNNs to automatic learn a representation of medical images and advantage of SCNN architecture for one-shot image learning that only needs a small dataset to train. We experimentally evaluated our model with combinations of MRI modalities. Our model achieve good results despite of using only small train set and dealing only with 2D image information.

In the future work, we are going to finish implementation remaining parts of the method that prediction will be done over three different orientations of lesion what reduce 3D information to 2D.

## References

- [1] Abraham, B., Nair, M.S.: Computer-aided classification of prostate cancer grade groups from MRI images using texture features and stacked sparse autoencoder. *Computerized Medical Imaging and Graphics*, 2018, vol. 69, pp. 60–68.
- [2] Bray, F., Ferlay, J., Soerjomataram, I., Siegel, R.L., Torre, L.A., Jemal, A.: Global cancer statistics 2018: GLOBOCAN estimates of incidence and mortality worldwide for 36 cancers in 185 countries. *CA: A Cancer Journal for Clinicians*, 2018, vol. 68, no. 6, pp. 394–424.
- [3] Koch, G.: Siamese Neural Networks for One-shot Image Recognition, 2011.
- [4] Le: Automated diagnosis of prostate cancer in multi-parametric MRI based on multimodal convolutional neural networks. *Phys. Med. Biol.*, 2018.
- [5] Lemaître, G.: *Computer-Aided Diagnosis for Prostate Cancer using Multi-Parametric Magnetic Resonance Imaging*. PhD thesis, 2016.
- [6] Liu, S., Zheng, H., Feng, Y., Li, W.: Prostate cancer diagnosis using deep learning with 3D multiparametric MRI, 2017, p. 1013428.
- [7] Mehrtaash, A., Sedghi, A., Ghafoorian, M., Taghipour, M., Clare, M., Kapur, T., Mousavi, P., Abolmaesumi, P., Fedorov, A., States, U., Image, D., Group, A.: Classification of Clinical Significance of MRI Prostate Findings Using 3D Convolutional Neural Networks, 2017.
- [8] Schröder, F.H., Hugosson, J., Roobol, M.J., Tammela, T.L., Ciatto, S., Nelen, V., Kwiatkowski, M., Lujan, M., Lilja, H., Zappa, M., Denis, L.J., Recker, F., Berenguer, A., Mänttinen, L., Bangma, C.H., Aus, G., Villers, A., Rebillard, X., van der Kwast, T., Blijenberg, B.G., Moss, S.M., de Koning, H.J., Auvinen, A.: Screening and Prostate-Cancer Mortality in a Randomized European Study. *New England Journal of Medicine*, 2009, vol. 360, no. 13, pp. 1320–1328.
- [9] Weng, Y.a.C.W.h.: Learning Deep Representations of Medical Images using Siamese CNNs with Application to Content-Based Image Retrieval, 2017, no. Figure 1, pp. 1–8.



## D Opis digitálnej časti práce

Evidenčné číslo práce v informačnom systéme: FIIT-5212-86031

Obsah digitálnej časti práce (archív ZIP):



Digitálna časť práce má veľkosť 5.1 GB, kvôli čomu je uložená v systéme G Suite for Education.

Názov odovzdaného archívu: PB\_prilohy\_digital\_FrantisekSefcik.zip