

Technická univerzita v Košiciach
Fakulta elektrotechniky a informatiky

**Automatická detekcia koronálnych dier a
aktívnych oblastí na Slnku**

Diplomová práca

2021

Bc. Martin Harman

Technická univerzita v Košiciach
Fakulta elektrotechniky a informatiky

**Automatická detekcia koronálnych dier a
aktívnych oblastí na Slnku**

Diplomová práca

Študijný program: Hospodárska informatika
Študijný odbor: Informatika
Školiace pracovisko: Katedra kybernetiky a umelej inteligencie (KKUI)
Školiteľ: doc. Ing. Peter Butka, PhD.
Konzultant: Ing. Viera Maslej Krešňáková
RNDr. Šimon Mackovjak, PhD.

Košice 2021

Bc. Martin Harman

Abstrakt v SJ

Kozmické počasie je fenomén, ktorý má obrovský dopad na naše každodenné životy v modernej spoločnosti. Slnko je strojcom života na Zemi, ale aj pôvodcom kozmického počasia. Vďaka najmodernejším satelitom na obežnej dráhe Zeme dokážeme monitorovať zdroje kozmického počasia - aktivitu Slnka v reálnom čase. Aby bolo možné tieto zdroje monitorovať automaticky, vytvorili sme model hlbokaj neurónovej siete na segmentáciu koronálnych dier a aktívnych oblastí. Náš model dosahuje výsledky, ktoré je možné porovnať so súčasnými metódami využívanými na detekciu štruktúr slnečnej koróny. Výstup nášho modelu je možné využiť na ďalšiu analýzu a výskum slnečnej koróny.

Kľúčové slová

Hlboké učenie, segmentácia, konvolučné neuronové siete, koronálne diery, aktívne oblasti

Abstrakt v AJ

Space weather is a phenomenon that has a huge impact on our daily lives in modern society. The sun is a maker of life on earth but also the cause of space weather. Thanks to modern spacecraft on Earth's orbit, we can monitor sources of space weather - the Sun in real-time. In order to monitor these resources, we have developed a model of a deep neural network for the segmentation of coronal holes and active regions. Our model provides results that can be compared which currently used segmentation models. The output of our model can be used for further analysis and research of the solar corona.

Kľúčové slová v AJ

Deep learning, segmentation, convolutional neural networks, coronal holes, active regions

TECHNICKÁ UNIVERZITA V KOŠICIACH
FAKULTA ELEKTROTECHNIKY A INFORMATIKY
Katedra kybernetiky a umelej inteligencie

ZADANIE DIPLOMOVEJ PRÁCE

Študijný odbor: **Informatika**
Študijný program: **Hospodárska informatika**

Názov práce:

Automatická detekcia koronálnych dier a aktívnych oblastí na Slnku
Automated detection of solar coronal holes and active regions

Študent: **Bc. Martin Harman**
Školiteľ: **doc. Ing. Peter Butka, PhD.**
Školiace pracovisko: **Katedra kybernetiky a umelej inteligencie**
Konzultant práce: **Ing. Viera Maslej Krešňáková, RNDr. Šimon Mackovjak, PhD.**
Pracovisko konzultanta: **Ústav experimentálnej fyziky SAV**

Pokyny na vypracovanie diplomovej práce:

1. Podať teoretický prehľad problematiky metód hlbokého učenia vhodných pre automatickú detekciu koronálnych dier a aktívnych oblastí na Slnku, s dôrazom na využitie konvolučných neuronových sietí pre segmentáciu.
2. Analyzovať a predspracovať dostupnú dátovú množinu, navrhnúť postupy pre detekciu koronálnych dier a aktívnych oblastí pomocou zvolených metód hlbokého učenia.
3. Realizovať a vyhodnotiť navrhnuté postupy detekcie na zvolenej množine dát.
4. Vypracovať dokumentáciu podľa pokynov katedry a vedúceho práce.

Jazyk, v ktorom sa práca vypracuje: slovenský
Termín pre odovzdanie práce: 23.04.2021
Dátum zadania diplomovej práce: 30.10.2020



prof. Ing. Liberios Vokorokos, PhD.
dekan fakulty

Čestné vyhlásenie

Vyhlasujem, že som diplomovú prácu vypracoval samostatne s použitím uvedenej odbornej literatúry.

Košice 20. 4. 2021

.....

Vlastnoručný podpis

Podakovanie

Chcel by som poďakovať doc. Ing. Peterovi Butkovi, PhD. za jeho odborné rady a pripomienky, ktoré pomohli k vzniku tejto práce. Taktiež by sa rád poďakoval mojej konzultantke Ing. Vierke Maslej Krešňákovej za jej sprevádzanie v oblasti hlbokého učenia a za jej rady a pripomienky pri navrhovaní modelu. Za uvedenie do problematiky kozmického počasia ďakujem RNDr. Šimonovi Mackovjakovi, PhD., ktorý mi pomáhal orientovať sa v tejto oblasti.

Obsah

Úvod	1
1 Hlboké učenie	3
1.1 Dopredné neuronové siete	4
1.2 Aktivačné funkcie	6
1.3 Optimalizácia	9
1.4 Chybová funkcia	10
1.5 Vyhodnocovacie metriky	11
1.6 Regularizácia	12
1.7 Konvolučné neuronové siete	14
1.8 Segmentácia obrazu	17
2 Slniečna koróna	20
2.1 Living with a Star	23
2.2 Slniečna koróna a hlboké učenie	25
2.3 Slniečna koróna – dostupné dáta	28
3 Segmentácia koronálnych dier a aktívnych oblastí	32
3.1 Použité technológie	33
3.2 Príprava dát	34
3.3 SCSS-Net	38
3.4 Post-processing	41
3.5 Vyhodnotenie výsledkov	42
3.5.1 Koronálne diery	43
3.5.2 Aktívne oblasti	45
4 Záver	50
Zoznam príloh	59

Zoznam obrázkov

1-1	Vennov diagram umelej inteligencie	4
1-2	Jednovrstvový perceptrón	5
1-3	Viacvrstvový perceptrón	6
1-4	Priebeh funkcie sigmoid	8
1-5	Priebeh funkcie ReLU	9
1-6	Ukážka pod/preučenia	13
1-7	Neurónová sieť po použití regularizačnej techniky dropout, ktorej následkom je vynechanie neurónu	14
1-8	Konvolúcia	16
1-9	Pridanie okraju – <i>Padding</i>	16
1-10	Max-pooling	16
1-11	Klasifikácia - Detekcia - Segmentácia	17
1-12	Ukážka snímky (vľavo) a segmentačnej masky (vpravo) zo súťaže ISBI-2012	18
1-13	U-Net architektúra	19
2-1	Aktívna oblasť (vľavo) a koronálna diera (vpravo) – SDO/AIA 193Å	21
2-2	Koronálna slučka – TRACE 171Å	22
2-3	Prístroj SDO/AIA	24
2-4	Ukážka snímok vyhotovených vo filtroch SDO/AIA	25
2-5	Priestorové zobrazenie výskytu AR (vľavo) a CH (vpravo)	27
2-6	Snímky slnka SDO/AIA 171Å (vľavo) a 193Å (vpravo) – 2. október 2020 23:00	29
2-7	Porovnanie anotácií koronálnych dier	31
3-1	Čiastočné zatmenie Slnka 11. Marca 2013	33
3-2	Predspracovanie a augmentácia dát	35
3-3	Rozdiel v intenzite jasu na snímkach z roku 2010 (vľavo) a z roku 2018 (vpravo) – SDO/AIA	36

3-4	SCSS-Net architektúra	38
3-5	Priebeh učenia	41
3-6	Post-processing	42
3-7	Porovnanie segmentácií koronálnych dier	46
3-8	Ukážka segmentácií koronálnych dier	47
3-9	Porovnanie segmentácií aktívnych oblastí	48
3-10	Ukážka segmentácií aktívnych oblastí	49

Zoznam tabuliek

3-1	Prehľad získaných anotácií CH	37
3-2	Prehľad získaných anotácií AR	38
3-3	Výsledky modelov za použitia rôznych optimalizačných funkcií	40
3-4	Výsledky modelov (CH) na validačnej množine	44
3-5	Výsledky modelov na testovacej množine zloženej zo SPoCA anotácií	44
3-6	Výsledky modelov na testovacej množine zloženej z Region Growth anotácií	45
3-7	Výsledky modelov (AR) na validačnej množine	46
3-8	Výsledky modelov na testovacej množine	48

Zoznam symbolov a skratiek

Adam Adaptive Moment Estimation

AI Artificial Intelligence

AIA Atmospheric Imaging Assembly

Angström (Å) dĺžková miera používaná v spektroskopii

API Application Programming Interface

AR Active Region

BCE Binary Cross-entropy

CE Cross-entropy

CH Coronal Hole

CNN Convolutional Neural Network

EUV Extreme Ultraviolet

FNN Feedforward Neural Network

HEK Heliophysics Event Knowledgebase

IoU Intersection over Union

LReLU Leaky Rectified Linear Unit

LWS Living with a Star

MLP Multilayer Perceptron

MSE Mean Squared Error

PReLU Parametric Rectified Linear Unit

ReLU Rectified Linear Unit

SCCS-Net Solar Coronal Structures Segmentation Network

SDO The Solar Dynamics Observatory

SGD Stochastic Gradient Descent

SOHO The Solar and Heliospheric Observatory

SPoCA Spatial Possibilistic Clustering Algorithm

Úvod

Slnko sa nachádza v strede našej slnečnej sústavy a je zdrojom všetkého života na Zemi. Prebieha v ňom obrovské množstvo fyzikálnych javov, niektoré z týchto javov majú vplyv nielen na Slnko samotne, ale aj na našu Zem. Všetky takéto javy ovplyvňujúce Zem a jej blízke okolie nazývame kozmické počasie. Kozmické počasie je ovplyvňované najmä slnečným vetrom, ktorý vzniká v koronálnych dierach a aktívnych oblastiach na Slnku. Aby sme dokázali lepšie porozumieť týmto solárnym javom potrebujeme ich prv správne identifikovať.

V spolupráci so RNDr. Šimonom Mackovjakom, PhD., ktorý pracuje na Oddelení kozmickej fyziky, Ústav experimentálnej fyziky Slovenskej Akadémie Vied sme navrhli proces automatickej detekcie aktívnych oblastí a koronálnych dier pomocou metód hlbokého učenia, nakoľko súčasné metódy segmentácie týchto oblastí zvyčajne využívajú klasické metódy strojového učenia ako zhľukovanie. Jednou z inšpirácií k vytvoreniu segmentačného modelu hľbokej neurónovej siete na segmentáciu aktívnych oblastí a koronálnych dier bola práca autorov [Illarionov and Tlatov \(2018\)](#), ktorá sa venovala segmentácií koronálnych dier. Snažili sme sa tak priniesť nové postupy, presnejší segmentačný model a venovať sa aj segmentácií aktívnych oblastí.

V tejto práci sme sa venovali teórií hlbokého učenia, predovšetkým metódam, ktoré sme využili pri vytváraní nášho modelu hľbokej neurónovej siete. Dôraz sme kládli na konvolučné neuronové siete a architektúru využívanú na segmentáciu snímok.

Nakoľko kozmické počasie nie je oblasť, ktorej sa venuje hospodárska informatika, tejto oblasti sme sa venovali len stručne. Podali sme prehľad o tom čo sú, kedy a ako vznikajú koronálne diery a aktívne oblasti a ako môžu ovplyvňovať život na Zemi a technológie na jej obežnej dráhe.

V analytickej časti sme sa bližšie pozreli na využitie strojového učenia v oblasti kozmického počasia. V tejto časti sme odhalili možné aplikácie hlbokého učenia,

ktoré ponúkajú voľne dostupné dáta zo satelitných meraní. Následne sme sa bližšie venovali prácam, ktoré sa venovali klasifikácií alebo segmentácií koronálnych dier a aktívnych oblastí. V nasledujúcej časti sa venujeme dostupným dátam, ktorými sú snímky Slnka a anotácie aktívnych oblastí a koronálnych dier. Neskôr sme popísali proces predspracovania týchto dát. Opísali sme architektúru našej konvolučnej neurónovej siete, ktorú sme nazvali SCSS-Net a vyhodnotili natrénované segmentačné modely.

1 Hlboké učenie

Hlboké učenie robí obrovské pokroky v riešení problémov, na ktoré tradičné metódy strojového učenia nestačia. Ukázalo sa, že metódy hlbokého učenia sú veľmi dobré pri objavovaní zložitých štruktúr vo viacrozmerných dátach, a sú preto využiteľné v mnohých oblastiach vedy, ekonomiky a vládneho sektoru. Hlboké učenie prinieslo prielom v oblasti rozpoznávania snímok pomocou konvolučných neurónových sietí. V oblasti spracovania prirodzeného jazyka nachádza hlboké učenie obrovské uplatnenie na klasifikáciu textu, rozpoznávania sentimentu a prekladu (Lecun et al., 2015).

V súčasnosti sa metódy hlbokého učenia tešia veľkej popularite hlavne v oblasti spracovania prirodzeného jazyka a počítačového videnia. Pojem hlboké učenie bol prvýkrát použitý autorom Rina Dechter v roku 1986 (Dechter, 1986). Prvý model neurónovej siete však bol predstavený autormi Warren McCulloch a Walter Pitts v roku 1943 (Schmidhuber, 2015).

V počiatkoch umelej inteligencie sa vedci zameriavali na riešenie problémov, ktoré sú náročné pre ľudí. Pri riešení takýchto problémov dosahovala umelá inteligencia úspech, pretože ich riešenie vedeli opísať formálnymi a matematickými pravidlami. Nakoniec sa ukázalo, že skutočnou výzvou umelej inteligencie sú úlohy, ktoré ľudia riešia intuitívne, automaticky a nepredstavujú tak pre nás náročnú úlohu.

Umelej inteligencii chýbala schopnosť extrahovať znalosti z pôvodných dát a zvyčajne sa rozhodovala pomocou vopred definovaných pravidiel. Túto schopnosť počítačom prinieslo až predstavenie strojového učenia. Algoritmy strojového učenia sa spoliehajú, že na vstupe dostanú sadu informácií, vďaka ktorým sa algoritmus naučí korelácie medzi jednotlivými informáciami. Tieto informácie tiež nazývame príznaky (angl. *features*). Výber vhodných príznakov má veľký dopad na výsledný algoritmus strojového učenia. Pre veľa úloh je až príliš náročné sa rozhodnúť, ktoré vlastnosti extrahovať. Na riešenie takýchto úloh sa preto začalo využívať hlboké učenie, vďaka ktorému sme schopní extrahovať komplexné príznaky na riešenie zložitých úloh (Goodfellow et al., 2016).

Obrázok 1-1 znázorňuje vzťah medzi umelou inteligenciou, strojovým učením a hlbokým učením. Tento vzťah ukazuje, že strojové učenie a hlboké učenie sú podmnožinou umelej inteligencie. Vývojom umelej inteligencie sa snažíme napodobiť ľudské správanie a myslenie. Pri strojovom učení zvyčajne využívame štatistické metódy a extrahované znalosti, ktoré slúžia na učenie algoritmu. O tom aké znalosti z dát extrahovať rozhoduje dátový vedec po dôkladnej analýze dát. Algoritmy hlbokého učenia však nepotrebujú vopred extrahované znalosti, pretože sa dokážu samé naučiť potrebné informácie zo vstupných dát.



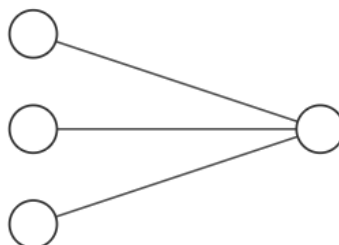
Obrázok 1-1 Vennov diagram umelej inteligencie

1.1 Dopredné neuronové siete

Dopredné neuronové siete sú zostavené z neurónov, ktoré sú usporiadané vo vrstvách. Prvú vrstvu siete nazývame vstupná (angl. *input layer*), poslednú vrstvu nazývame výstupná (angl. *output layer*) a vrstvy, ktoré sa nachádzajú medzi vstupom a výstupom nazývame skryté (angl. *hidden layer*) vrstvy (Svozil et al., 1997).

Neurónová sieť zložená iba z dvoch vrstiev (vstupu a výstupu), je najjednoduchšia forma neurónovej siete (viď. Obrázok 1-2). Takúto sieť označujeme ako jedno-

vrstvomý perceptrón (angl. *singlelayer perceptron*) a je kľúčovým komponentom vo viacvrstvomých dopredných sieťach (Šarūnas Raudys, 1998).



Vstupná vrstva $\in \mathbb{R}^3$ Výstupná vrstva $\in \mathbb{R}^1$

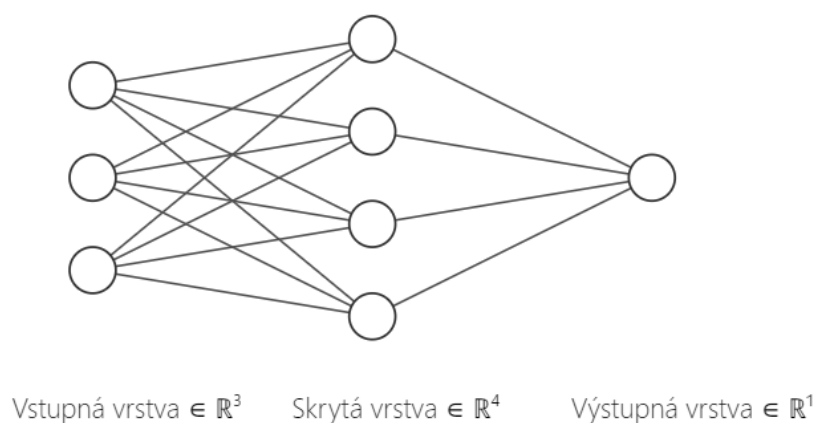
Obrázok 1 – 2 Jednovrstvomý perceptrón

Jednovrstvomý perceptrón bol vyvinutý vedcami Rumelhart et al. (1985) inšpirovaný prácou vedcov McCulloch and Pitts (1943). Jednovrstvomý perceptrón je algoritmus binárneho klasifikátora nazývaný prahová funkcia (angl. *threshold function*), ktorý z jedného alebo viacerých binárnych vstupov vytvorí jeden binárny výstup (Nielsen, 2015). Tento perceptrón je vyjadrený vzťahom 1.1, kde x je vstup, $f(x)$ je binárny výstup a $\sum_j w_j x_j$ je vážená suma vstupu.

$$f(x) \begin{cases} 0 & \text{ak } \sum_j w_j x_j \leq \text{prah} \\ 1 & \text{ak } \sum_j w_j x_j > \text{prah} \end{cases} \quad (1.1)$$

Zložením viacerých jednovrstvomých perceptrónov vytvoríme viacvrstvomý perceptrón (angl. *multilayer perceptron*, skr. MLP). MLP je nazývaný aj ako dopredná neurónová sieť (angl. *feedforward neural network*, skr. FFN), pretože informácie v nich prúdia cez funkciu f v jednom smere od vstupu x cez skryté vrstvy, ktoré definujú funkciu f , až po výstup y a nazývame ich sieťami, pretože sú zvyčajne zložené z viacerých rôznych funkcií. Model neurónovej siete zvykne byť reprezentovaný usmerneným acyklickým grafom popisujúcim ako sú prepojené funkcie v neurónovej sieti (Goodfellow et al., 2016). Na Obrázku 1–3 je znázornený viacvrstvomý perceptrón – dopredná neurónová sieť, kde vidíme neuróny v jednotlivých vrstvách, ktoré

sú prepojené medzi sebou. Neuróny v jednej vrstve sú prepojené s neurónmi vo vrstve nasledujúcej a tieto prepojenia medzi jednotlivými neurónmi sú charakterizované váhovým koeficientom, ktorý značí dôležitosť tohto prepojenia (Svozil et al., 1997).



Obrázok 1 – 3 Viacvrstvový perceptrón

1.2 Aktivačné funkcie

Dopredné neuronové siete predstavili koncept skrytých vrstiev vyžadujúce výber aktivačnej funkcie, ktorá sa použije na výpočet hodnôt skrytej vrstvy (Goodfellow et al., 2016). Aktivačné funkcie, niekedy nazývané aj prenosové funkcie, definujú výstup neurónu v rozsahu hodnôt. Medzi najzákladnejšie aktivačné funkcie patria kroková alebo prahová aktivačná funkcia, *piecewise* aktivačná funkcia, logistická a hyperbolická aktivačná funkcia (Rios et al., 2020). V neuronových sieťach aktivačná funkcia definuje ako sa vstup transformuje na výstup. Vo viacvrstvových hlbokých neuronových sieťach zvyčajne využívame jednu aktivačnú funkciu v skrytých vrstvách, a inú na vrstve výstupnej. V skrytých vrstvách sa najčastejšie využíva ReLU, logistická aktivačná funkcia alebo hyperbolický tangens. Vo výstupnej vrstve vyberáme aktivačnú funkciu podľa toho, aký je náš očakávaný výstup z neuronovej siete:

- regresia – lineárna aktivačná funkcia,
- binárna klasifikácia – sigmoidálna aktivačná funkcia,
- klasifikácia do viacerých tried – softmax aktivačná funkcia.

V tejto kapitole sa budeme bližšie venovať sigmoidálnej funkcii a ReLU, pretože práve tieto funkcie využívame v našom modeli, ktorému sa budeme venovať v Kapitole [3]. Vo všetkých skrytých vrstvách využívame aktivačnú funkciu ReLU a na výstupnej vrstve využívame sigmoid, pretože sa rozhodujeme, či daný pixel patrí alebo nepatrí do danej triedy.

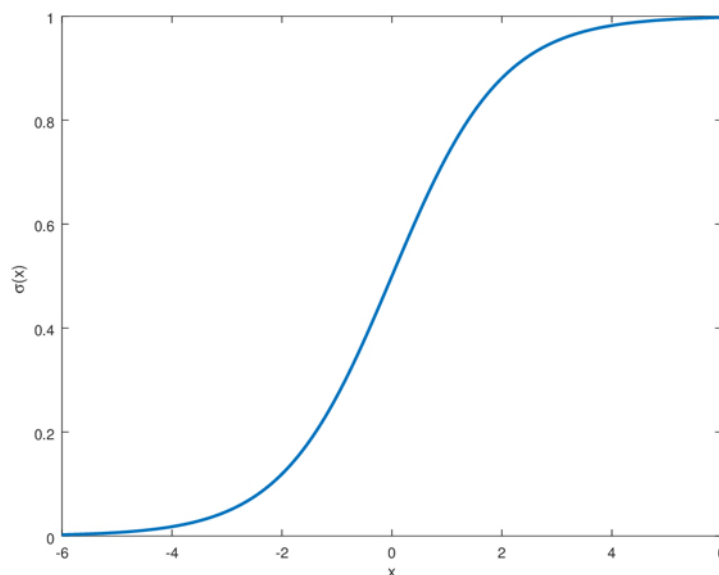
Sigmoidálna funkcia

Sigmoidálna funkcia je vďaka jej nelinearite jedna z najpoužívanejších funkcií v dopredných neurónových sieťach. Je to diferencovateľná funkcia, ktorá je definovaná pre všetky reálne vstupné hodnoty a má všade pozitívnu deriváciu. Vykazuje dostatočný stupeň hladkosti a je tiež vhodným rozšírením limitujúcich nelinearít, ktoré boli v minulosti využívané v neurónových sieťach (Han and Moraga, 1995). Nazývame ju aj ako logistická funkcia a je známa svojím tvarom písmena S zobrazeným na Obrázku [1-4]. Logistická funkcia je vyjadrená vzťahom [1.2] a jej hodnoty jej výstupu sa pohybujú medzi 0 a 1. Vďaka tomu sa najčastejšie využíva ako aktivačná funkcia na výstupnej vrstve pre binárnu klasifikáciu.

$$f(x) = \frac{1}{1 + e^{-x}} \quad (1.2)$$

Rektifikovaná lineárna jednotka

Rektifikovaná lineárna jednotka (skr. ReLU), vyjadrená vzťahom [1.3] patrí medzi najobľúbenejšie funkcie využívané na skrytej vrstve, a to vďaka jej jednoduchosti a výpočtovej nenáročnosti. Ako môžeme vidieť na Obrázku [1-5], tak ReLU nadobúda nulovú hodnotu pre vstup $x < 0$, ale pre vstup $x \geq 0$ nadobúda nezmenené hodnoty vstupu. ReLU umožňuje sieti ľahko získať reprezentácie pri nízkej náročnosti na



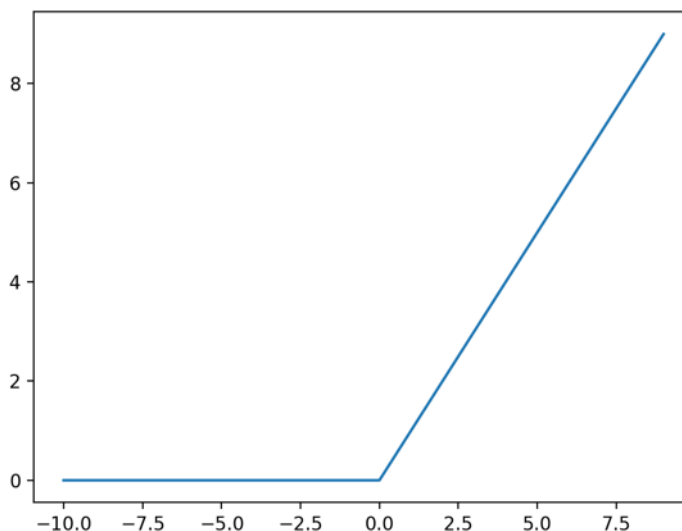
Obrázok 1 – 4 Priebek funkcie sigmoid

výpočtový výkon vďaka tomu, že nevyužíva exponenciálne funkcie. Napriek experimentálnym výsledkom, ktoré ukazujú, že nulové hodnoty môžu pomôcť pri učení s učiteľom (angl. *supervised learning*), môže ReLU spôsobovať problém miznúcich gradientov (Glorot et al., 2011).

$$f(x) = \max(0, x) \quad (1.3)$$

Problém miznúcich gradientov môžeme vyriešiť využitím optimalizačnej funkcie Leaky ReLU (skr. LReLU), ktorá je vyjadrená vzťahom 1.4 a na rozdiel od klasickej ReLU funkcie, LReLU nadobúda malé nenulové hodnoty (Maas et al., 2013). Obdobou LReLU funkcie je Parametrická ReLU funkcia (skr. PReLU) vyjadrená vzťahom $f(x) = \max(ax, x)$, kde $a \leq 1$ je učiteľný parameter. V prípade ak $a = 0$ získame tým ReLU funkciu a ak $a = 0.01$ dostaneme LReLU (He et al., 2015).

$$f(x) \begin{cases} x & \text{ak } x \geq 0 \\ 0.01x & \text{ak } x < 0 \end{cases} \quad (1.4)$$



Obrázok 1 – 5 Priebeh funkcie ReLU

1.3 Optimalizácia

Väčšina algoritmov hlbokého učenia vyžaduje nejakú formu optimalizácie. Pod optimalizáciou chápeme úlohu minimalizovať alebo maximalizovať funkciu $f(x)$, úpravou x . Maximalizácia môže byť dosiahnutá pomocou minimalizačného algoritmu minimalizáciou $-f(x)$ (Goodfellow et al., 2016). V hlbokom učení sa optimalizáciou neurónovej siete snažíme minimalizovať chybu pri učení úpravou atribútov neurónovej siete, ako napríklad rýchlosť učenia. Na optimalizáciu neurónovej siete sa najčastejšie využívajú metódy gradientového zostupu, ako napríklad Stochastický gradient zostupu (skr. SGD) alebo SGD s momentom (Robbins 2007; Kiefer et al., 1952). Pre prípad, kedy je gradient v jednom smere oveľa väčší než v inom sa využívajú metódy s adaptívnou rýchlosťou učenia, ako napríklad Adagrad (Duchi et al., 2011), RMSProp (Tieleman and Hinton, 2012) alebo Adam (Kingma and Ba, 2017).

Počas tréovania našej hlbkej neurónovej siete sme najlepšie výsledky dostali ak sme využili optimalizačný algoritmus Adam, preto si v nasledujúcej časti bližšie popíšeme práve tento optimalizačný algoritmus.

Adam (*Adaptive Moment Estimation*) kombinuje výhody metód AdaGrad a RMSProp. Je to algoritmus pre optimalizáciu funkcií na základe gradientu prvého rádu a je založený na adaptívnych odhadoch momentov nižšieho rádu. Táto metóda je výpočtovo efektívna, má malé nároky na pamäť a je vhodná na riešenie problémov, ktoré vyžadujú veľké množstvo dát alebo parametrov. Aktualizácia parametrov optimizéra Adam je daná vzťahmi 1.5 až 1.9, kde máme parametre $w^{(t)}$ a chybovú funkciu $L^{(t)}$, pričom t je index iterácie. (Kingma and Ba, 2017).

$$m_w^{(t+1)} \leftarrow \beta_1 m_w^{(t)} + (1 - \beta_1) \nabla_w L^{(t)} \quad (1.5)$$

$$v_w^{(t+1)} \leftarrow \beta_2 v_w^{(t)} + (1 - \beta_2) (\nabla_w L^{(t)})^2 \quad (1.6)$$

$$\hat{m}_w = \frac{m_w^{(t+1)}}{1 - \beta_1^{t+1}} \quad (1.7)$$

$$\hat{v}_w = \frac{v_w^{(t+1)}}{1 - \beta_2^{t+1}} \quad (1.8)$$

$$w^{(t+1)} \leftarrow w^{(t)} - \eta \frac{\hat{m}_w}{\sqrt{\hat{v}_w + \epsilon}}, \quad (1.9)$$

kde $\beta_1 = 0.9$ je momentová konštanta, $\beta_2 = 0.999$ je tlmiaci faktor a aby sa zabránilo deleniu nulou slúži malé číslo η .

1.4 Chybová funkcia

Chybová funkcia je funkcia, ktorú sa snažíme minimalizovať. Takúto funkciu nazývame ako cieľová funkcia alebo kritérium. V rámci minimalizácie funkcie sa takáto funkcia nazýva taktiež nákladová, stratová alebo chybová funkcia (Goodfellow et al., 2016). Na to aby sme vypočítali akú chybu dosahuje model počas optimalizačného procesu je potrebné zvoliť vhodnú chybovú funkciu. Samotný výber funkcie môže byť problém sám o sebe, pretože nie vždy je ľahké vytvoriť funkciu, ktorá meria presne to, čo chceme. Často je potrebné urobiť kompromis medzi tým, čo chceme, čo môžeme merať a čo môžeme efektívne optimalizovať. Najčastejšie sa používa niekoľko základných funkcií (Reed and Marks, 1998):

- priemerná štvorcová chyba (angl. *mean squared error*, skr. MSE) pri problémoch s aproximáciou (regresiou),
- krížová entropia (angl. *cross-entropy*, skr. CE) pri klasifikačných úlohách.

Keďže našou úlohou je segmentácia snímok, čo je vlastne len klasifikácia pixelov, tak sme ako chybovú funkciu využívali krížovú entropiu. Bližšie sa jej budeme venovať v nasledujúcej časti.

Krížová entropia bola motivovaná adaptívnym algoritmom na odhad pravdepodobnosti zriedkavých udalostí v zložitých sieťach, ktorý zahŕňa minimalizáciu odchýlok (De Boer et al. 2005). Krížová entropia je mierou rozdielu medzi dvoma rozdeleniami pravdepodobnosti pre danú náhodnú premennú alebo množinu udalostí a je vyjadrená vzťahom 1.10. Pre prípad binárnej klasifikácie sa využíva binárna krížová entropia (angl. *binary cross-entropy*, skr. BCE) vyjadrená vzťahom 1.11, kde y reprezentuje pravú hodnotu a \hat{y} reprezentuje predikovanú hodnotu (Murphy, 2012).

$$\text{CE} = - \sum_i^M y_i \log(\hat{y}_i) \quad (1.10)$$

$$\text{BCE} = - (y \log(\hat{y}) + (1 - y) \log(1 - \hat{y})), \quad (1.11)$$

1.5 Vyhodnocovacie metriky

Na vyhodnotenie modelov sa zvyknú používať metriky, ktoré nám pomáhajú jednoducho porovnať výsledky viacerých modelov medzi sebou. Medzi najčastejšie používanú metriky na vyhodnotenie klasifikačných algoritmov patrí celková úspešnosť (angl. *accuracy*), ktorá vyjadruje pomer správne klasifikovaných tried oproti všetkým klasifikovaným triedam. Pre úlohu segmentácie však táto metrika nie je práve najvhodnejšia, pretože nám nehovorí nič o nesprávne klasifikovaných pixeloch.

Segmentačné modely sa najčastejšie vyhodnocujú pomocou Jaccardovho indexu (angl. *Jaccard index*) (Legendre and Legendre, 2012) a Dice koeficientu (Sorensen,

1948). Jaccardov index je označovaný ako „priesečník cez zjednotenie“ (angl. *Intersection Over Union*, skr. IoU), pretože IoU vyjadruje prekrytie predikovanej oblasti so skutočnou oblasťou vydelený zjednotením oboch oblastí. Dice koeficient, ktorý je známy v dátovej analytike ako F1 skóre vyjadruje podobnosť medzi dvoma súbormi údajov. IoU a DICE sú vyjadrené vzťahmi:

$$\text{IoU} = \frac{\text{TP}}{\text{TP} + \text{FP} + \text{FN}} \quad (1.12)$$

$$\text{Dice} = \frac{2\text{TP}}{2\text{TP} + \text{FP} + \text{FN}} \quad (1.13)$$

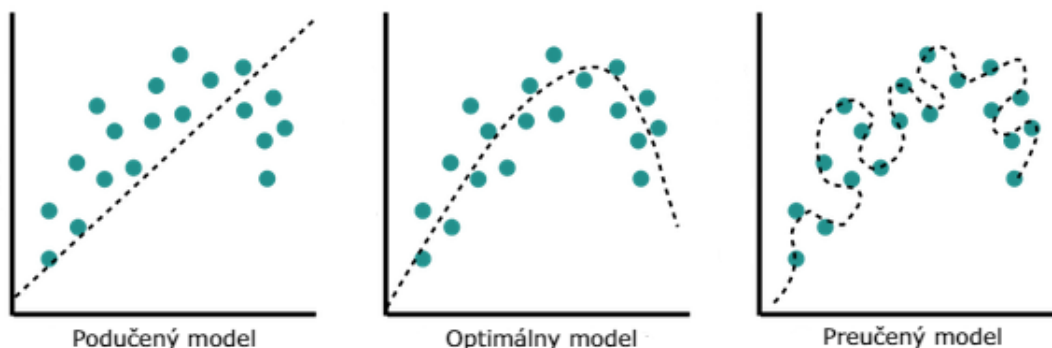
kde:

- TP – true positive – všetky pixely, ktoré boli klasifikované do správnej triedy,
- FP – false positive - všetky pixely, ktoré boli nesprávne označené, že patria do danej triedy,
- FN – false negative – všetky pixely, ktoré boli nesprávne označené, že do danej triedy nepatria.

1.6 Regularizácia

Skôr ako si popíšeme regularizačné techniky, tak si zavedieme pojmy tréningová množina, validačná množina a testovacia množina a vysvetlíme ako boli využité v tejto práci. Pod pojmom tréningová množina chápeme súbor dát, ktoré slúžia na tréning modelu. Ako validačnú množinu berieme malú časť dát z tréningovej množiny, ktorú sme nepoužili na tréning modelu, ale na vyhodnocovanie procesu učenia. Testovacia množina slúži ako nezávislá časť dát, na ktorej vyhodnocujeme natréningovaný model. Schopnosť dosahovať dobré výsledky na neznámych dátach sa nazýva generalizácia. V strojovom učení je jednou z najväčších výziev vytvorenie algoritmu, ktorý dosahuje dobré výsledky nielen na tréningových dátach, ale aj na dátach nových. Veľa stratégií v strojovom učení je navrhnutých na redukciu validačnej chyby, niekedy aj za cenu zvýšenia tréningovej chyby. Tieto stratégie sú označované ako regularizácia.

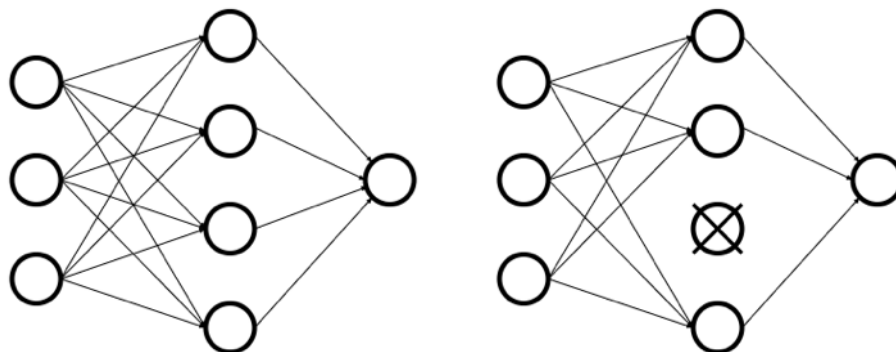
V strojovom učení sa snažíme aby bola zovšeobecnená chyba (angl. *generalization error*) čo najmenšia. Počas učenia neurónovej siete môže dôjsť k podučeniu (angl. *underfitting*), ku ktorému dochádza ak model nedosahuje dostatočne nízku chybu na tréningových dátach. V prípade klasifikácie to má za následok nesprávne zaradenie príkladov do tried. K preučeniu (angl. *overfitting*) dochádza ak je príliš veľký rozdiel medzi tréningovou a testovacou chybou, teda ak je testovacia chyba výrazne väčšia než tréningová chyba. V takomto prípade model dokáže veľmi presne klasifikovať príklady, ktoré už pozná alebo sú im príliš podobné, ale nedokáže správne zaradiť nové príklady (Goodfellow et al., 2016). Na Obrázku 1–6 je znázornený príklad podučenia na ľavej časti grafu kde chyba ešte nie je dostatočne nízka a preučenia na pravej časti grafu, kde chyba je dostatočne nízka, ale rozdiel medzi tréningovou a testovacou chybou je príliš veľký. K najčastejším formám regularizácie neurónových sietí patrí L1 a L2 norma, dropout, augmentácia dát a včasné zastavenie.



Obrázok 1–6 Ukážka pod/preučenia

Dropout je silná regularizačná stratégia, ktorú možno považovať za jednu z najlepších metód, ak chceme zabrániť preučeniu modelu. Je efektívnejší než iné regularizačné stratégie a môže byť kombinovaný s inými metódami regularizácie. Medzi výhody dropout metódy patrí to, že je výpočtovo nenáročný a nie je limitovaný typom neurónovej siete (Goodfellow et al., 2016). Funkcia dropout zabezpečuje vynechanie alebo vypadnutie neurónov v neurónovej sieti. Na Obrázku 1–7 je zobrazené, že vynechaním neurónu docielime jeho dočasné vymazanie a prerušenie spojení s inými

neurónmi. Vynechanie neurónu zvyčajne prebieha náhodne na základe stanovenej pravdepodobnosti (Srivastava et al., 2014).



Obrázok 1–7 Neurónová sieť po použití regularizačnej techniky dropout, ktorej následkom je vynechanie neurónu

1.7 Konvolučné neuronové siete

Medzi najpopulárnejšie hlboké neuronové siete patria konvolučné neuronové siete (angl. *Convolutional Neural Network*, skr CNN), ktoré dosahujú vynikajúce výsledky vo oblasti počítačového videnia. Aj napriek tomu, že konvolučné neuronové siete boli dlho známe a dosahovali dobré výsledky, tak sa tieto modely sa najviac preslávili vďaka sieti AlexNet (Krizhevsky et al., 2012), ktorá zaznamenala obrovský úspech v klasifikácii snímok pochádzajúcich z ImageNet databázy, ktorá obsahuje viac než milión snímok rozdelených do 1000 tried (Lecun et al., 2015). Základnou myšlienkou konvolučných neuronových sietí je navrhnúť také riešenie, ktoré nám umožní prehĺbiť sieť za použitia menšieho počtu parametrov. Tento cieľ sa zvyčajne dosahuje zmenšením veľkosti snímky na vstupe siete, ale hlavne v skrytých vrstvách siete pomocou konvolučných a vzorkovacích vrstiev (Aghdam and Heravi, 2017).

Konvolučné neuronové siete sú špeciálnym druhom neuronových sietí, ktoré sú zamerané na spracovanie dát, ktoré majú formu mriežky. Tieto dáta môžu byť jednorozmerné napr. časové rady a text, dvojrozmerné napr. obrázky alebo viacrozmerné. Označenie „konvolučná“ vychádza zo špeciálnej lineárnej matematickej ope-

rácie dvoch funkcií nazývanej konvolúcia, ktorá sa zvykne vyjadrovať vzťahom:

$$s(t) = (x * w)(t), \quad (1.14)$$

kde znak $*$ predstavuje konvolúciu funkcií x a w . Prvá funkcia x je označovaná ako vstup a druhá funkcia w je označovaná ako kernel resp. konvolučný filter. Výstup konvolúcie funkcií x a w sa nazýva mapa príznakov (angl. *feature map*). Typická konvolučná neurónová sieť je zložená z troch fáz. V prvej fáze konvolúcie vytvárajú lineárne aktivačné funkcie, ktoré následne v druhej fáze, nazývanej fáza detekcie putujú do nelineárnej aktivačnej funkcie napr. ReLU a v tretej fáze sa používa plne prepojená vrstva, ktorá slúži na klasifikáciu (Goodfellow et al., 2016).

- **Konvolučná vrstva** (angl. *convolutional layer*) je základná súčasť CNN, ktorej základom je matematická operácia – konvolúcia. Na Obrázku 1–8 máme vstupnú oblasť (vľavo), ktorá reprezentuje binárny obrázok, konvolučný filter (v strede) a výstup konvolúcie (vpravo). Pri aplikovaní konvolučného filtra určujeme aj vertikálny a horizontálny posun (angl. *stride*), tie určujú o koľko bodov resp. pixelov sa bude konvolučný filter posúvať. Veľkosť vstupu, veľkosť a posun filtra ovplyvňujú veľkosť výstupu. V našom prípade sme na vstup o veľkosti 5×5 aplikovali 3×3 konvolučný filter s posunom 1 vo vertikálnom aj horizontálnom smere, to malo za dôsledok, že výstup konvolúcie mal veľkosť 3×3 . Ak chceme aby si výstup po konvolúciách zachoval rovnakú veľkosť ako vstup, môžeme vstupnému obrázku pridať okraj v podobe núl. Pri použití *padding* sa môže stred konvolučného filtra ocitnúť v rohu snímky, čo môže pomôcť modelu lepšie zachytiť príznaky na okraji snímky. Výsledok aplikácie nulového okraja (angl. *zero-padding*) je zobrazený na Obrázku 1–9.
- **Združovacia vrstva (vzorkovacia)** (angl. *pooling layer*) sa využíva na redukovanie veľkosti výstupu a nahrádza ho súhrnnou štatistikou, napr. priemerom alebo maximálnymi hodnotami (Goodfellow et al., 2016). Na Obrázku 1–10 máme vstup o veľkosti 4×4 , ktorý zredukujeme pomocou 2×2 max-poolingu

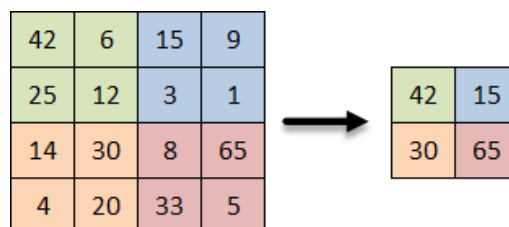
$$\begin{array}{|c|c|c|c|c|} \hline 1 & 1 & 1 & 0 & 0 \\ \hline 0 & 1 & 1 & 1 & 0 \\ \hline 0 & 0 & 1 & 1 & 1 \\ \hline 0 & 0 & 1 & 1 & 0 \\ \hline 0 & 1 & 1 & 0 & 0 \\ \hline \end{array}
 \cdot
 \begin{array}{|c|c|c|} \hline 1 & 0 & 1 \\ \hline 0 & 1 & 0 \\ \hline 1 & 0 & 1 \\ \hline \end{array}
 =
 \begin{array}{|c|c|c|} \hline 4 & 3 & 4 \\ \hline 2 & 4 & 3 \\ \hline 2 & 3 & 4 \\ \hline \end{array}$$

Obrázok 1–8 Konvolúcia

0	0	0	0	0	0	0
0	1	1	1	0	0	0
0	0	1	1	1	0	0
0	0	0	1	1	1	0
0	0	0	1	1	0	0
0	0	1	1	0	0	0
0	0	0	0	0	0	0

Obrázok 1–9 Pridanie okraju – *Padding*

na veľkosť 2×2 , týmto zmenšíme náročnosť siete, pričom zachované dôležité informácie. Hlavným cieľom združovacej vrstvy je znížiť rozmer mapy príznakov, preto sa nazýva aj ako vzorkovanie. Združovanie sa uplatňuje na každej mape príznakov osobitne. To znamená, že ak výstup konvolučnej vrstvy má 50 máp príznakov, tak sa operácia združovania použije na každú z týchto máp osobitne a vytvorí sa tak 50 nových máp príznakov, ktoré budú mať zredukované rozmery (Aghdam and Heravi 2017).



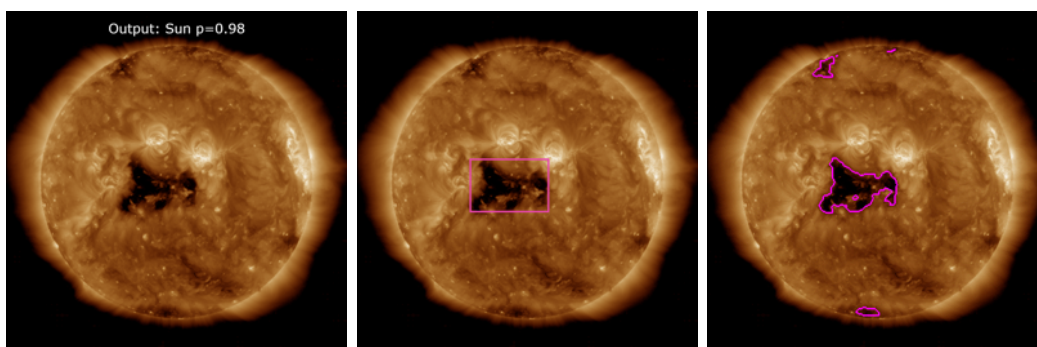
Obrázok 1–10 Max-pooling

- **Plne prepojená vrstva** (angl. *fully connected layer*) slúži ako dopredná neurónová sieť, ktorá sa využíva ako výstupná vrstva konvolučných neurónových sietí. Vstupom do tejto vrstvy je matica transformovaná na jednorozmerný vektor. Ten je vstupom pre aktivačnú funkciu, zvyčajne *sigmoid* alebo *softmax* podľa toho, či ide o binárnu klasifikáciu alebo klasifikáciu viacerých tried (Goodfellow et al., 2016).

1.8 Segmentácia obrazu

Počítačové videnie bolo tradične jednou z najaktívnejších oblastí výskumu aplikácií hlbokého učenia, pretože videnie je úloha, ktorá je pre ľudí jednoduchá, ale pre počítače náročná (Ballard et al., 1983). Oblasť počítačového videnia a rozoznávania snímok je veľmi obrovská a zahŕňa širokú škálu spôsobov spracovania snímok a úžasnú rozmanitosť aplikácií (Goodfellow et al., 2016). Poznáme tri základné metódy rozpoznávania snímok, znázorene na Obrázku 1–11:

- **Klasifikácia** – označí do akej triedy patrí daná snímka.
- **Detekcia** – detegovaný objekt ohraničí obdĺžnikom.
- **Segmentácia** – určí, ktoré pixely patria danému objektu.



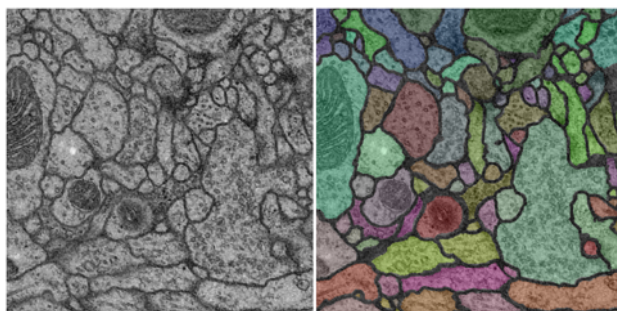
Obrázok 1 – 11 Klasifikácia - Detekcia - Segmentácia

V minulosti sa na segmentáciu snímok využívalo prahovanie (Sahoo et al., 1988), zhlukovanie (Hartigan and Wong, 1979) alebo detekcia hrán (Canny, 1986). V sú-

časnosti medzí najúspešnejšie techniky segmentácie snímok patria metódy hlbokého učenia, konkrétne konvolučné neuronové siete (Lecun et al. 2015), ktoré boli opísané v podkapitole 1.7.

Architektúra U-Net

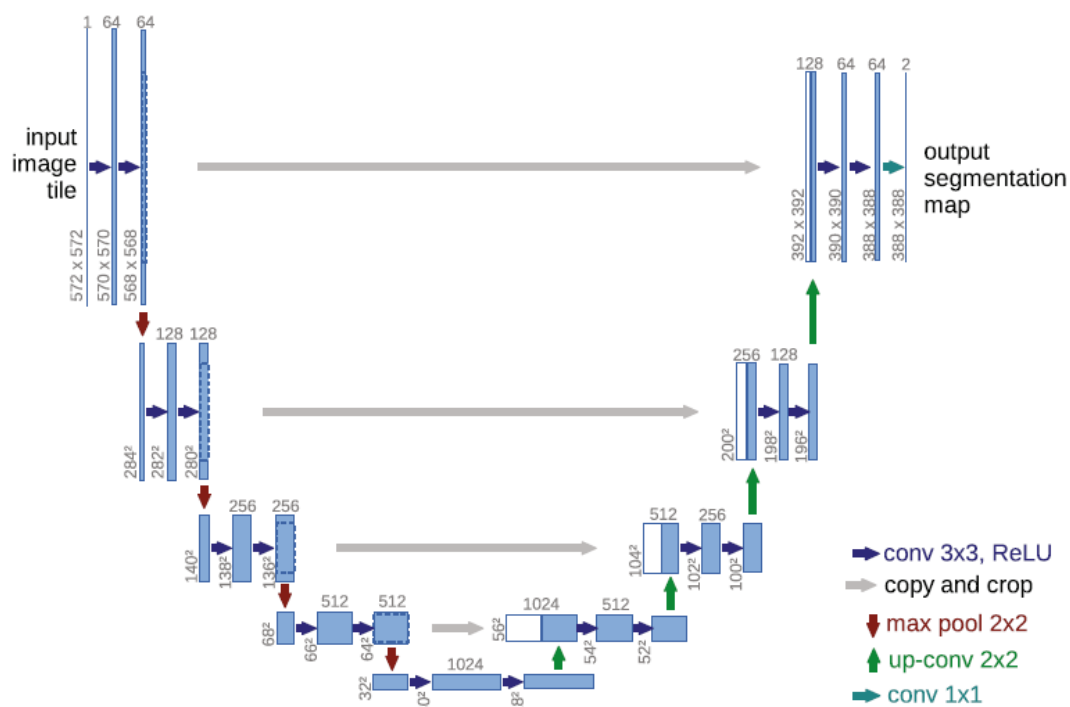
U-Net bola predstavená autormi (Ronneberger et al. 2015) a zaznamenala veľmi dobré výsledky pri segmentácii biomedicínskych snímok. V roku 2012 vyhrala *ISBI Challenge* – súťaž zameranú na automatickú segmentáciu neuronálnych štruktúr (viď. Obrázok 1–12).



Obrázok 1 – 12 Ukážka snímky (vľavo) a segmentačnej masky (vpravo) zo súťaže ISBI-2012

Táto architektúra je úpravou a rozšírením plne prepojenej konvulčnej neurónovej siete predstavenej autormi (Long et al. 2014). U-Net bola modifikovaná, aby vytvárala presnejšie segmentácie za použitia menšieho počtu trénovacích snímok. Táto neuronová sieť nemá žiadne plne prepojené vrstvy a segmentačné mapy obsahujú iba tie pixely, pre ktoré je na vstupnom obrázku k dispozícii celý kontext. Na predikciu pixelov v okrajovej oblasti snímky sa chýbajúci kontext extrapoluje zrkadlením vstupnej snímky.

Architektúra U-Net je zobrazená na Obrázku 1–13 a je zložená z enkódera (ľava časť) a dekódera (prava časť). Enkodér je typickou časťou konvulčných neurónových sietí a pozostáva zo vstupu a niekoľkých aplikácií dvoch konvulčných vrstiev s veľkosťou filtra 3×3 pixelov. Následne zredukovala veľkosti snímky pomocou funkcie *max-pooling* s veľkosťou okna 2×2 pixelov. Dekodér tento proces opakuje v opač-



Obrázok 1 – 13 U-Net architektúra

nom smere, kde namiesto zmenšovania veľkosti snímky pomocou *max-pooling*, sieť snímku postupne zväčšuje na pôvodnú veľkosť pomocou funkcie *up-sampling* s veľkosťou okna 2×2 pixelov. Na prepojenie správnych veľkostí snímok a extrahovaných príznakov v jednotlivých úrovniach siete je použitá funkcia *concatenate*, ktorá slúži ako prepojenie medzi enkodérom a dekodérom. Na výstupe je použitá konvolučná vrstva s veľkosťou filtra 1×1 , ktorá slúži na klasifikáciu (Ronneberger et al., 2015). Práve táto sieť sa stala odrazovým mostíkom pre náš výskum segmentácie slnečných štruktúr.

2 Slnečná koróna

Kozmické počasie predstavuje zmeny vo vesmírnom prostredí medzi Slnkom a Zemou. Štúdie kozmického počasia sa zaoberajú javmi, ktoré ovplyvňujú život na Zemi a systémy na obežnej dráhe Zeme. Dokonca aj počasie na Zemi, od povrchu planéty smerom do vesmíru, je ovplyvňované Slnkom. A tak kozmické a aj pozemské počasie je určované malými zmenami, ktoré Slnko podstúpi počas svojho slnečného cyklu (*Space Weather Prediction Center*, 2021).

Medzi oblasti , na ktoré má kozmické počasie najväčší vplyv patria:

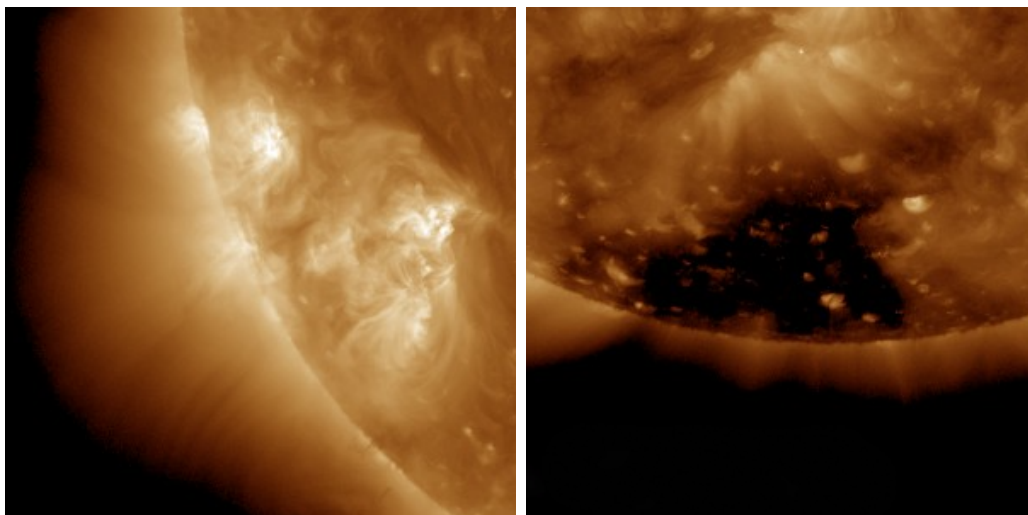
- klíma Zeme,
- elektrická sieť,
- GPS systémy,
- satelitná komunikácia.

Slnečná fyzika je rozvíjajúca sa vedecká disciplína integrujúca štúdie o premenlivosti Slnka, okolitej heliosfére a klimatickom prostredí. Slnečná fyzika, nazývaná aj heliofyzika je oblasť fyziky, ktorá sa zaoberá Slnkom a slnečnou aktivitou. Za posledných pár storočí naše chápanie toho, ako Slnko riadi vesmírne počasie a podnebie na Zemi a iných planétach, napredovalo čoraz rýchlejšie (*Schrijver and Siscoe*, 2010). Vo vonkajších atmosférach na Slnku, môžu vznikáť rôzne prírodné javy, ktoré nazývame solárne javy, medzi ktoré patrí:

- slnečný vietor,
- slnečné erupcie,
- výrony koronálnej hmoty,
- slnečné škvrny,
- koronálne diery.

Tieto solárne javy môžu spôsobovať problémy našim moderným technológiám, ktoré sú citlivé na rôzne výkyvy kozmického počasia. Zmeny v ionosfére počas geomagnetických búrok narúšajú vysokofrekvenčnú rádiovú komunikáciu a navigáciu v systéme GPS a energetické častice vo forme slnečného vetra poškodzujú kritickú elektroniku satelitov na obežnej dráhe Zeme. Napríklad v roku 1983 zapríčinili geomagnetické indukované prúdy kolaps Hydro-Québeckej elektrickej siete, čo spôsobilo poruchu transformátora a 9 hodinový výpadok elektriny. Túto udalosť zapríčinila geomagnetická búrka, ktorá vznikla v dôsledku výronu koronálnej hmoty.

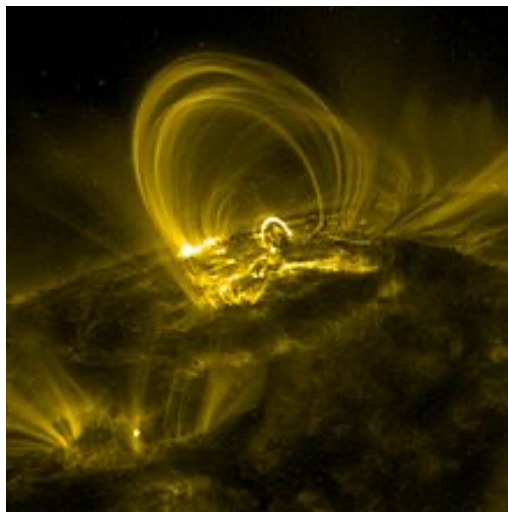
Slnko prechádza pravidelnými zmenami alebo cyklami vysokej a nízkej aktivity, ktoré sa opakujú približne každých 11 rokov. Počas týchto cyklov dochádza k tzv. solárnemu minimu a solárnemu maximu. Počas solárneho minima sa na slnku objavuje najmenej slnečných škvŕn a počas solárneho maxima je zas počet slnečných škvŕn najväčší (Garner, 2015).



Obrázok 2 – 1 Aktívna oblasť (vľavo) a koronálna diera (vpravo) – SDO/AIA 193Å

Koronálne diery, zobrazené na Obrázku 2-1 sa na extrémne ultrafialových snímkach javia ako tmavé oblasti v slnečnej koróne. Javia sa ako tmavé škvrny, pretože sú to chladnejšie a menej husté v porovnaní s ich okolím. Sú to oblasti s otvoreným unipolárnym magnetickým polom, ktoré umožňujú slnečnému vetru ľahšie uniknúť do otvoreného vesmíru. Dlhotrvajúce koronálne diery sú zdrojom vysokorýchlostných prúdov slnečného vetra. Označenie „koronálna” odkazuje na vonkajšiu vrstvu slnečnej atmosféry, ktorú nazývame koróna.

Koronálne diery môžu mať trvanie až 27 dní a sú najčastejšie a najstabilnejšie na severom a južnom póle Slnka, ale môžu sa rozšíriť aj do nižších slnečných šírok. Priemerná životnosť polárnych koronálnych dier je približne 6 až 10 rotácií Slnka. Niekedy sa časť koronálnej diery oddelí od koronálnej diery, ktorá vznikla na póle a stane sa samostatnou. Najčastejšie sa vyskytujú v rokoch po solárnom maxime resp. počas rokov solárneho minima (Rušin and Rybanský 1990; *Coronal Holes* 2021).



Obrázok 2-2 Koronálna slučka – TRACE 171Å

Aktívne oblasti, zobrazené na Obrázku 2-1 sú oblasti so silným magnetickým polom, ktoré môže byť až 1000 krát silnejšie ako priemerné magnetické pole Slnka. Najčastejšie ich môžeme spozorovať počas vrcholu slnečného cyklu, kedy je magnetické pole Slnka narušené. Aktívne oblasti sa na röntgenových a ultrafialo-

vých snímkach Slnka sa javia ako oblasti s vysokou intenzitou. Silné magnetické pole okolo aktívnej oblasti uvoľňuje intenzívne výbuchy energie, ktoré majú formu vysokoenergetických fotónov a častíc. Okolo aktívnych oblastí sa často objavujú koronálne slučky zobrazené na Obrázku 2-2 (van Driel-Gesztelyi and Green, 2015; *Active Regions on the Sun*, 2021).

2.1 Living with a Star

Living with a Star (skr. LWS) je program NASA, ktorý ma za cieľ poskytnúť vedecké poznatky v oblasti slnečnej fyziky a jej aspektov, ktoré môžu mať vplyv na život a spoločnosť. Konečným cieľom je vyvinúť schopnosť predikovať neustále sa meniacu aktivitu Slnka. Aktuálne NASA misie vrámci programu LWS:

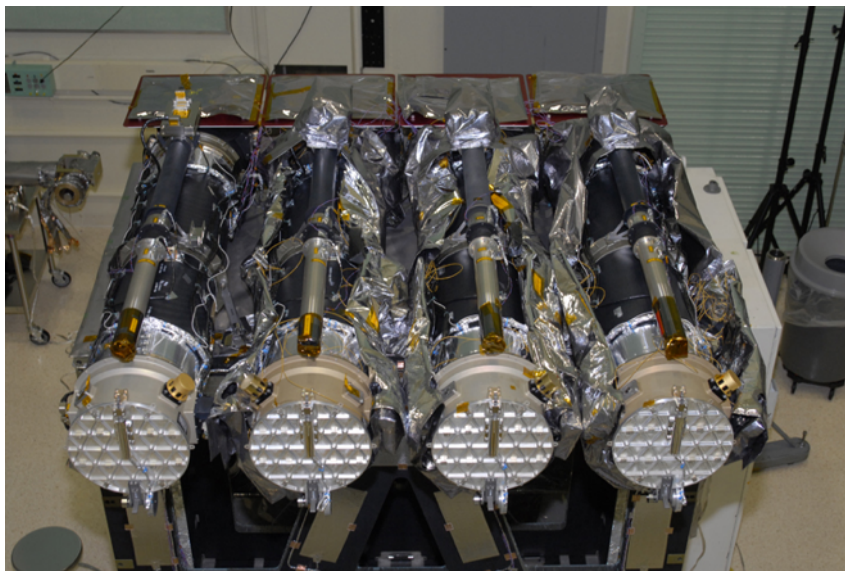
- The Solar Dynamics Observatory (skr. SDO)
- Van Allen Probes (skr. VAP)
- Parker Solar Probe (skr. PSP)
- Solar Orbiter (skr. SolO) – v spolupráci s ESA

The Solar Dynamics Observatory je prvá misia, ktorá bola spustená vrámci programu LWS vo februári 2010. Cieľom SDO je porozumieť slnečnej aktivite a následne predikovať tie slnečné udalosti, ktoré ovplyvňujú život na zemi a technológie (Pesnell et al., 2011). Tento cieľ sa snaží dosiahnuť pomocou zodpovedania otázok:

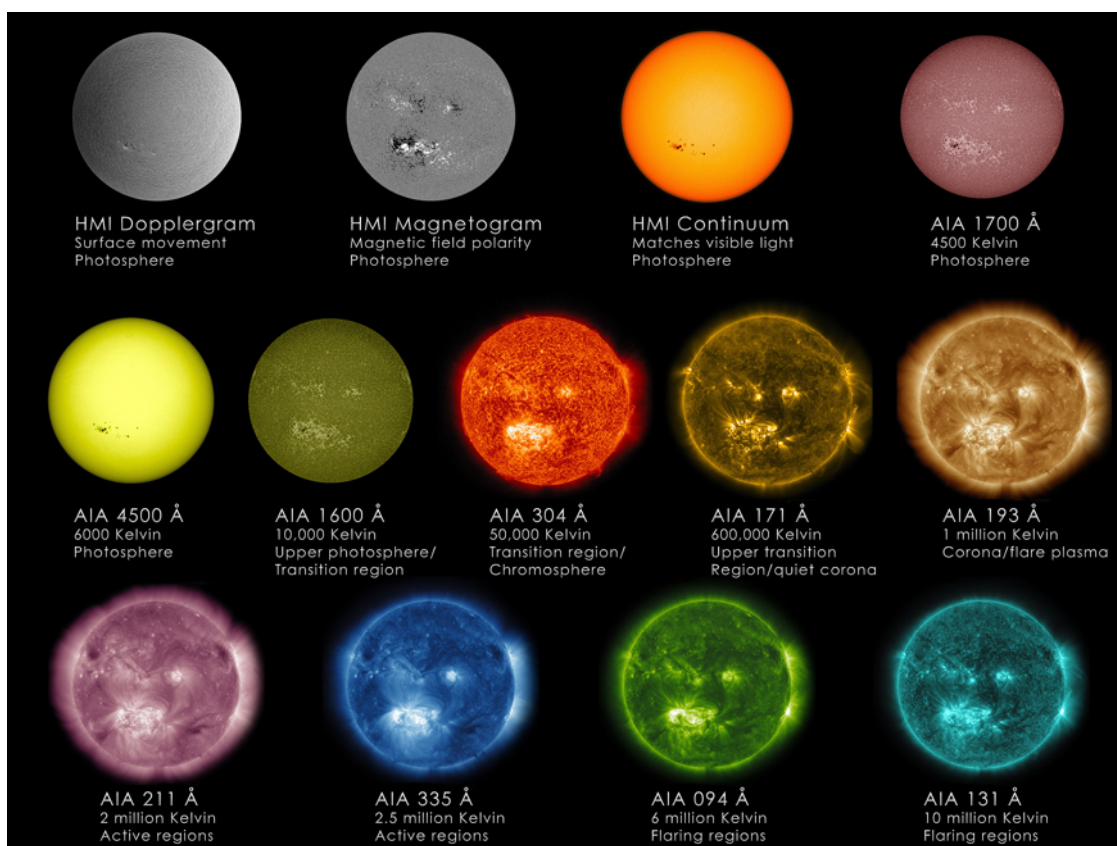
- Ako je generované a štrukturované magnetické pole Slnka?
- Ako uchovaná magnetická energia je konvertovaná a vypustená do heliosféry a priestoru zeme vo forme slnečného vetru, energických častíc a slnečného žiarenia?

Na palube SDO sa nachádza niekoľko výskumných zariadení:

- **Atmospheric Imaging Assembly** (skr. AIA) – vyhotovuje snímky slnka každých 12 sekúnd v 10 vlnových dĺžkach (94Å, 131Å, 171Å, 193Å, 211Å, 304Å, 335Å, 1600Å, 1700Å, 4500Å). Prístroj AIA je zobrazený na Obrázku [2-3](#) a snímky Slnka v jednotlivých vlnových dĺžkach sú zobrazené na Obrázku [2-4](#)
- **EUV Variability Experiment** (skr. EVE) – meria spektrum extrémne ultrafialového žiarenia Slnka s vysokou presnosťou.
- **Helioseismic and Magnetic Imager** (skr. HMI) – meria slnečné magnetické pole. Rozširuje možnosti SOHO/MDI prístroja, pokrytím celého disku pri vyššom priestorovom rozlíšení a novými schopnosťami vektorového magnetogramu.



Obrázok 2 – 3 Prístroj SDO/AIA



Obrázok 2–4 Ukážka snímok vyhotovených vo filtroch SDO/AIA

2.2 Slnečná koróna a hlboké učenie

Podľa [Camporeale \(2019\)](#) pre oblasť kozmického počasia, nie je strojové učenie a hlavné neuronové siete neznáma oblasť. Prvé pokusy o vytvorenie predikčných modelov neurónových sietí siahajú až do obdobia 90-tých rokov. Avšak aj napriek tomu, že neuronové siete boli využívané aj v tejto oblasti, tak konvolučné neurónové siete boli v oblasti kozmického počasia využité minimálne. Oblasť kozmického počasia má veľký potenciál na aplikovanie metód strojového a hlbokého učenia, pretože má všetky potrebné prvky, ktoré sú potrebné pre úspešnú aplikáciu strojového učenia. K dispozícii máme obrovské množstvo dát, ktoré boli zhromaždené počas niekoľkých desaťročí vesmírnych misií. V mnohých odvetviach sú dáta strážené a nedostupné, ale v oblasti kozmického počasia je väčšina dát dostupná pre širokú verejnosť. Na-

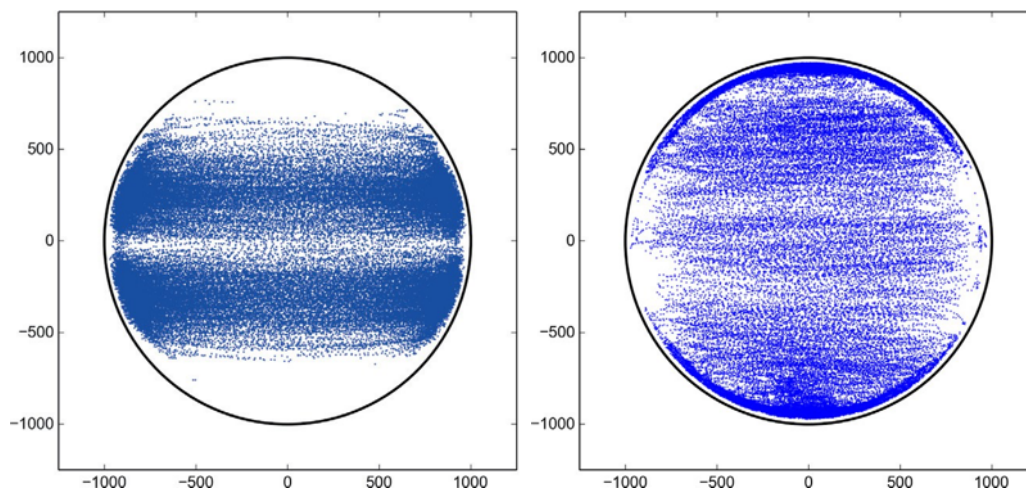
príklad satelity Advanced Composition Explorer (skr. ACE) a Wind and the Deep Space Climate Observatory (skr. DSCOVR) poskytujú údaje o plazme v blízkosti prvého Lagrangeovho bodu (skr. L1) s niekoľkými časovými rozlíšeniami, z ktorých sa niektoré datujú až 20 rokov dozadu. Satelity The Solar and Heliospheric Observatory (skr. SOHO) a SDO, opísané v podkapitole 2.1 poskytujú magnetogramy, koronogramy a snímky slnka v rôznych vlnových dĺžkach. Keďže sa jedná o oblasť kozmického počasia tak asi nebude žiadnym prekvapením, že najobľúbenejšou úlohou pre aplikáciu strojového učenia je predikcia a to hlavne:

- predpoveď geomagnetických indexov,
- predikcia relativistických elektrónov na geosynchrónnej obežnej dráhe Zeme,
- predpoveď slnečných erupcií,
- predpovedanie vývoja výronov koronálnej hmoty.

Práca Schuh et al. (2016) predstavila komplexný prehľad solárnych javov a snímok zachytených satelitom SDO, ktoré sú verejne dostupné prostredníctvom Heliophysics Event Knowledgebase (skr. HEK) modulu. Autori vytvorili početné štatistiky a exploratívne analýzy, ktoré poskytujú cenné kontextové informácie pre lepšie pochopenie dát uložených v HEK – úložisku informácií o solárnych javoch, ku ktorým je možné pristúpiť pomocou aplikačného rozhrania (angl. *application programming interface*, skr. API) (Hurlburt et al., 2012). Solárne javy zozbierané z HEK databázy sa odohrali v časovom období od začiatku roka 2010 až po koniec roka 2014. Medzi zozbieranými solárnymi javmi boli:

- aktívne oblasti,
- koronálne diery,
- slnečné škvrny,
- slnečné erupcie a ďalšie.

Na Obrázku 2–5 je vykreslená priestorová poloha aktívnych oblastí a koronálnych dier na základe stredov anotácií pochádzajúcich z algoritmu SPoCA, ktorý je stručne popísaný v kapitole 2.3. Tento obrázok nám vizuálne dokazuje, že sa koronálne diery najčastejšie vyskytujú pólach Slnka.



Obrázok 2–5 Priestorové zobrazenie výskytu AR (vľavo) a CH (vpravo)

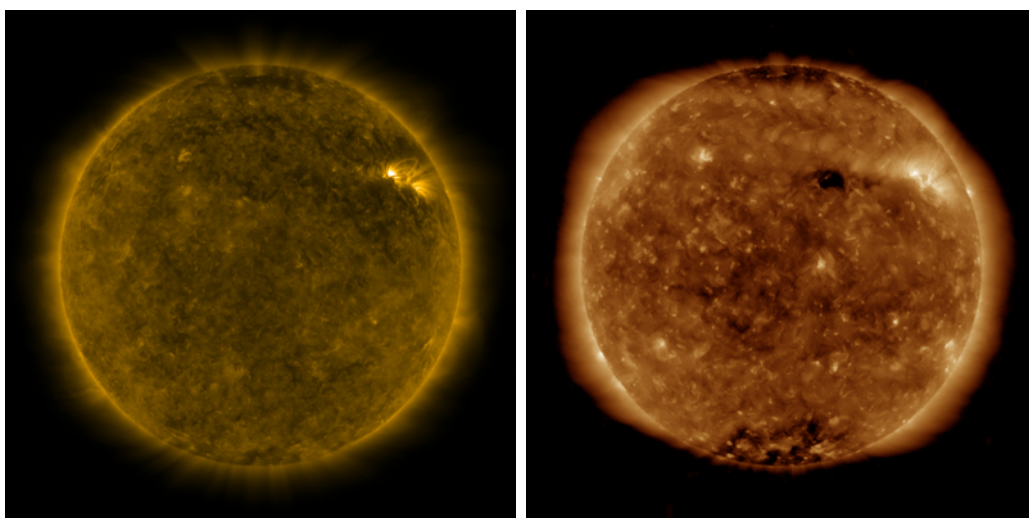
Segmentácií aktívnych oblastí, koronálnych dier alebo iných solárnych javov sa venovalo hneď niekoľko prác, avšak iba málo z nich využilo na segmentáciu hlboké neuronové siete. Najčastejšie sú využívané prahovacie alebo zhlukovacie metódy. Nižšie si popíšeme niekoľko prác venujúcich sa detekciou solárnych udalostí, z ktorých sa najväčšej pozornosti dostáva hlavne detekciou koronálnych dier. Práca autorov [Henney and Harvey \(2005\)](#) sa venovala automatickej detekcii koronálnych dier za pomoci spektroheliogramov a fotosférických magnetogramov. Vytvorili algoritmus automatickej detekcie, ktorý využíva morfológickú analýzu obrazu, prahovanie a vyhladzovanie na odhad polohy, hraníc a polaritu koronálnych dier. Autori [Krista and Gallagher \(2009\)](#) využili extrémne ultrafialové a röntgenové snímky vyhotovené prístrojmi STEREO, SOHO a Hinode na identifikáciu koronálnych dier pomocou prahovania intenzity založenej na histograme. Najväčšou limitáciou tejto metódy, bola detekcia koronálnych dier v blízkosti okraja, kedy dochádzalo k nesprávnemu určeniu hraníc oblasti. Upravená verzia algoritmu SPoCA a snímky z nástroja SDO

boli využitá autormi [Delouille et al. \(2018\)](#) na segmentáciu koronálnych dier. Cieľom ich práce bolo vylepšiť algoritmus SPoCA a vytvoriť klasifikačný model, ktorý by rozpoznával, či sa na snímke nachádza koronálna diera alebo filament. Na klasifikáciu využili niekoľko algoritmov strojového učenia ako napríklad rozhodovacie stromy, k-nn a SVM. Oblasť kozmického počasia sa venuje častejšie koronálnym dieram než aktívnym oblastiam, pričom sú najčastejšie využívané klasické metódy strojového učenia. Vďaka tomu sa nám vyskytuje príležitosť využiť konvolučné neuronové siete na automatickú detekciu týchto oblastí. Segmentáciou koronálnych dier pomocou konvolučných neurónových sietí sa však venovali aj autori [Illarionov and Tlatov \(2018\)](#). Autori v práci sami uvádzajú a dúfajú, že ich práca bude motiváciou pre pokračovanie výskumu využitia neurónových sietí na analýzu Slnkových snímok. Preto ich práca pre nás slúžila ako motivácia priniesť lepší model neurónovej siete na detekciu nielen koronálnych dier, ale aj aktívnych oblastí. Na tréning modelu využili snímky Slnka vo vlnovej dĺžke 193Å vyhotovené satelitom SDO. Ako zdroj anotácií im slúžil algoritmus, ktorý nazvali region growth. Ich neurónová sieť bola založená na architektúre U-Net predstavenej v podkapitole [1.7](#) a dávajú dôraz na to, že výsledky dosiahnuté ich neurónovou sieťou je možné zlepšiť dôraznejšou úpravou siete. Na vyhodnotenie modelu využili testovaciu množinu, ktorá pozostávala zo snímok Slnka z roku 2017. Model vyhodnocovali pomocou DICE koeficientu, ktorého priemerná hodnota na testovacej množine bola na úrovni 81%. S iným prístupom prišli autori [Jatla et al. \(2020\)](#), ktorých cieľom je predikcia geomagnetických búrok. K tomuto cieľu sa chcú priblížiť pomocou viac-modelovej segmentácie koronálnych dier extrémne ultrafialové snímky (angl. *extreme ultraviolet*, skr. EUV) a magnetické mapy.

2.3 Slnčná koróna – dostupné dáta

Pre natréningovanie modelu neurónovej siete slúžiaceho na segmentáciu obrazových dát potrebujeme 2 typy dát. Prvým typom sú snímky, na ktorých sa nachádza

objekt určený na segmentáciu a druhým typom je informácia, kde sa daný objekt na konkrétnej snímke nachádza. Touto informáciou môže byť sústava súradníc, ktorá určuje hranice daného objektu alebo snímka vo forme čierno-bielej binárnej masky, ktorá určuje či daný pixel je (1, resp. biela farba) alebo nie je súčasťou (0, resp. čierna farba) objektu.



Obrázok 2 – 6 Snímky slnka SDO/AIA 171Å (vľavo) a 193Å (vpravo) – 2. október 2020 23:00

V nasledujúcej časti stručne popíšeme ako sme získali tieto dáta:

1. **Snímky** pochádzajú z programu SDO/AIA opísaného v podkapitole [2.1](#) Pre našu úlohu sme vybrali snímky slnka v dvoch rozličných vlnových dĺžkach, ktorými sú 171Å a 193Å. Ako môžeme vidieť na obrázku [2–6](#) na snímkach vo vlnovej dĺžke 171Å je najväčší kontrast medzi aktívnymi oblasťami a povrchom slnka čo, ale neplatí pre koronálne diery, ktoré sú však veľmi dobré viditeľné na snímkach vo vlnovej dĺžke 193Å a preto sme rozhodli že:

- 171Å - budú slúžiť na segmentáciu aktívnych oblastí (angl. *active region*, skr. AR),
- 193Å - budú slúžiť na segmentáciu koronálnych dier (angl. *coronal holes*, skr. CH).

V našej práci sme využívali snímky v rozlíšení 4096x4096 vo formáte JPEG 2000 (Taubman and Marcellin, 2002) a PNG snímky získané z HEK API (Hurlburt et al., 2012). Snímky Slnka je však možné získať aj z iných zdrojov¹

2. **Anotácie** aktívnych oblastí a koronálnych dier sú dostupné zvyčajne vo forme súradníc, ktoré ohraničujú danú oblasť, tieto súradnice sme následne spracovali a vytvorili z nich binárne masky. Pre AR máme k dispozícii iba 2 zdroje anotácií, zatiaľ čo anotácií CH máme viacero zdrojov.

Zdroje anotácií:

- SPoCA-suite² (AR a CH) – Zo SPoCA-suite sme získali anotácie pre AR aj CH. Tento nástroj využíva tri typy fuzzy zhlukovacích algoritmov: Fuzzy C-means (FCM) a jej regularizovanú verziu nazývanú Possiblistic C-means (PCM) a Spatial Possiblistic Clustering Algorithm (SPOCA) (Delouille et al., 2012).
- Vlastné (AR a CH) – Tieto anotácie vznikli v rámci anotačného projektu Zooniverse (Simpson et al., 2014), ktorý bol súčasťou bakalárskej práce (Urbán, 2020). Anotovanie pozostávalo z ručného označovania AR a CH.
- CHIMERA³ (CH) – Algoritmus Chimera bol vytvorený na detekciu koronálnych dier a prvýkrát predstavený v publikácii (Garton, Tadhg M. et al., 2018). Tento algoritmus funguje na princípe segmentácie intenzity medzi tromi vlnovými rozsahmi 171Å, 193Å a 211Å, ide o tzv. „multi-thermal intensity segmentation”.
- Region Growth⁴ (CH) – Tento algoritmus bol taktiež vytvorený len na detekciu koronálnych dier a bol predstavený v publikácii (Tlatov et al., 2018).

¹Ďalšie zdroje snímok: <https://sdo.gsfc.nasa.gov/data/> https://sdowww.lmsal.com/suntoday_v2/ <http://jsoc.stanford.edu/>

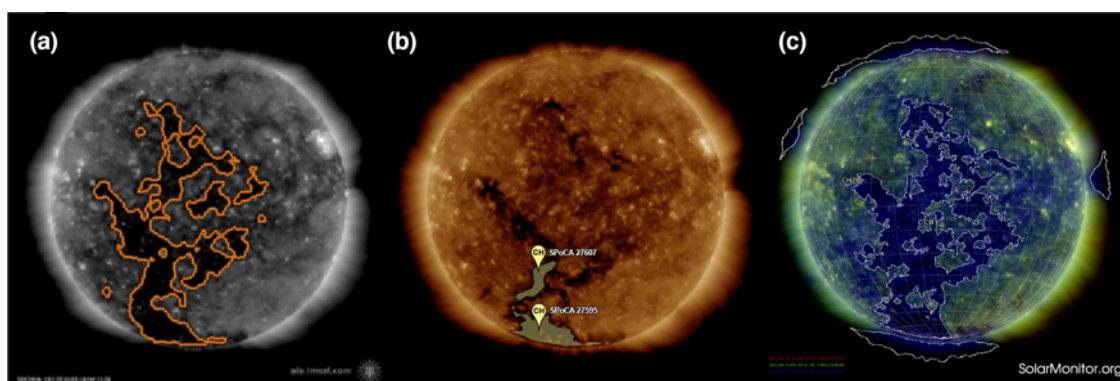
²Dostupné online na <https://helioviewer.org/> a pomocou Sunpy knižnice a HEK API v jazyku Python

³Dostupné online na <https://solarmonitor.org>

⁴Dostupné online na <https://observethesun.com/>

(2014). Region Growth funguje na princípe poloautomatického spracovania SDO/AIA 193Å snímok.

Na snímke 2–7 (A–region growth, B–SPoCA, C–CHIMERA) autorov Illarionov and Tlatov (2018) alebo na snímke 3–7 (B–SPoCA, C–region growth, D–CHIMERA) môžeme vidieť porovnanie anotácií rôznych zdrojov, z ktorých SPoCA má často nepresné anotácie, ktoré nezahŕňujú celú koronálnu diery alebo zahŕňujú oblasť, ktorá už nieje koronálna diera. Anotácie CHIMERA zahŕňujú cele koronálne diery, ale zároveň aj menej tmavé oblasti nazývané filamente. Region growth anotácie sú výsledkom poloautomatickej segmentácie a môžeme ich vyhodnotiť ako najlepšie.

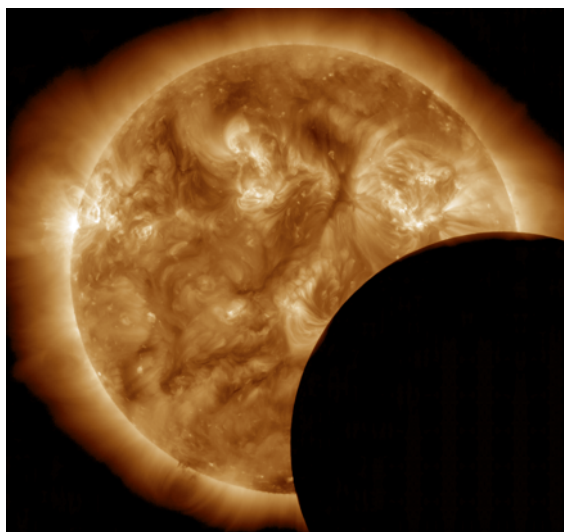


Obrázok 2–7 Porovnanie anotácií koronálnych dier

3 Segmentácia koronálnych dier a aktívnych oblastí

Pre vytvorenie úspešného klasifikačného, segmentačného alebo napríklad predikčného modelu je dôležité stanoviť si cieľ, ktorý chceme pomocou daného modelu dosiahnuť. V našom prípade ide o vytvorenie segmentačného modelu pomocou metód hlbokého učenia, ktorý prinesie lepšie segmentácie než doteraz využívaný SPoCA algoritmus. Cieľom nie je vytvoriť len model na segmentáciu konkrétnych udalosti (koronárnych dier či aktívnych oblastí), ale zároveň priniesť akýsi návod počnúc od vytvorenia segmentácii v Zooniverse, ktorým sa venovala bakalárska práca [Urbán \(2020\)](#) až po automatickú segmentáciu rôznych aj menej špecifických štruktúr. Komplexným cieľom našej výskumnej skupiny je teda priniesť podrobný návod ako segmentovať rozličné štruktúry slnečnej koróny. Naš prístup budeme testovať na dvoch rozdielnych udalostiach, aktívne oblasti špecifické jasným žiarením a na druhej strane koronálne diery špecifické pravé tmavými oblasťami. Architektúra modelu by tak mala byť univerzálna aj pre iné špecifické segmentačné úlohy.

Jediná limitácia, ktorú v tejto aj v budúcej práci vidíme sú dáta. Preto veľmi dôležitým krokom je zoznámenie sa z dátami, ktoré sme predstavili v kapitole [2.3](#) a následné ich príprava do podoby, ktorú vyžaduje trénovaný model na vstupe. Proces prípravy dát zvyčajne vyžaduje ich očistenie od nepotrebných prvkov, v našom prípade sme odstránili poškodené snímky alebo snímky kde došlo k zakrytiu Mesiacom, ako môžeme vidieť na Obrázku [3-1](#). Súčasťou prípravy dát bolo aj ich predspracovanie a augmentácia, tieto procesy popíšeme v podkapitole [3.2](#). Nasledujúcim krokom je modelovanie, v ktorom vytvoríme a natrénujeme model a vyhodnotíme ho na základe určených kritérií. Po vyhodnotení modelu môžeme pokračovať nasadením modelu alebo sa vrátiť späť k predspracovaniu dát alebo modelovaniu. Takýto postup je označovaný ako CRISP-DM ([Paralic, 2003](#)) a je rozdelený do 6 fáz: pochopenie cieľa, pochopenie dát, príprava dát, modelovanie, vyhodnotenie, nasadenie.



Obrázok 3 – 1 Čiastočné zatmenie Slnka 11. Marca 2013

3.1 Použité technológie

Na vypracovanie praktickej časti tejto práce sme vyžili open-source programovací jazyk **Python**, ktorý má veľkú podporu pre hlboké učenie, analýzu dát, ale aj pre vedu. Iba pre oblasť snečnej fyziky existuje viac než 28 modulov resp. knižníc (Burrell et al., 2018). Využité knižnice:

- **Tensorflow**⁵ je open-source platforma vyvinutá spoločnosťou Google. Je to knižnica určená primárne na numerické výpočty a na vývoj hlbokých neurónových sietí. Tensorflow podporuje tréning neurónových sietí na viacerých CPU aj GPU.
- **Keras**⁶ sa stal pevnou súčasťou Tensorflow s príchodom verzie Tensorflow 2.0. Keras je intuitívna knižnica určená pre vývoj neurónových sietí, ktorá bola vyvinutá aby umožnila rýchly a jednoduchý vývoj a prototypovanie.
- **Astropy**⁷ je výsledkom snahy vytvoriť jednotnú knižnicu pre astrofyziku a astronómiu. Je to open-source knižnica vyvinutá pre jazyk Python, ktorá obsa-

⁵<https://www.tensorflow.org/>

⁶<https://keras.io/>

⁷<https://www.astropy.org/>

huje rôzne nástroje a funkcie používané v bežných astronomických nástrojoch (Price-Whelan et al., 2018; Robitaille et al., 2013).

- **Sunpy**⁸ je komunitou vyvíjaná open-source knižnica určená pre solárnu fyziku. Pretože Sunpy sa zaoberá astrofyzikálnymi konceptmi, je vývoj tejto knižnice úzko spojený s vývojom knižnice Astropy (The SunPy Community and Barnes, 2020).
- **OpenCV**⁹ je multiplatformová open-source knižnica určená pre oblasť počítačového videnia a strojového učenia. Obsahuje viac než 2500 optimalizovaných algoritmov. Je využívaná hlavne na manipuláciu s videom a snímkami.
- **Pillow**¹⁰ je knižnica pre jazyk Python určená na manipuláciu so snímkami jej hlavná výhoda oproti OpenCV spočíva v používateľskej prístupnosti.

3.2 Príprava dát

Prípravu dát, sme rozdelili do 2 krokov a to fixné zmeny, ktoré popíšeme v časti predspracovanie dát a augmentáciu dát, teda rozširovanie dátovej množiny augmentačnými technikami, ktorým sa budeme venovať v oddiele augmentácia dát.

Predspracovanie dát

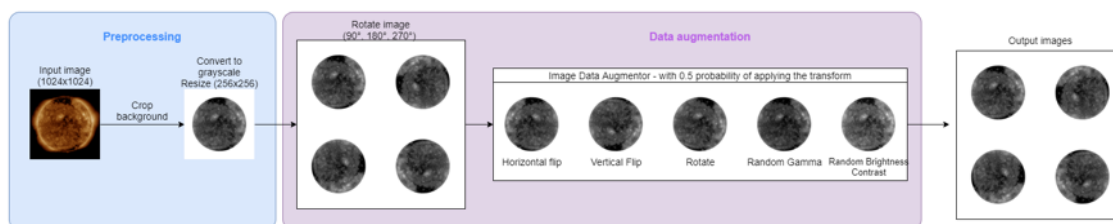
Proces predspracovania dát pre snímky 193Å (CH) obsahoval jeden krok navyše než pre snímky 171Å (AR). Tento krok spočíval v odstránení čierneho pozadia, ktoré spôsobovalo problémy pri segmentácii prevažne v oblasti pólův, pretože koronálne diery, ktoré sú charakterizované čiernymi pixelmi a splývali tak s čiernym pozadím. Na tento problém už poukázali (Illarionov and Tlatov, 2018) a po otestovaní sa nám prvotne potvrdil rovnaký výsledok. Tento problém sa nám neskôr podarilo eliminovať ďalšou úpravou siete, ale rozhodli sme sa zachovať tento proces predspracovania

⁸<https://sunpy.org/>

⁹<https://opencv.org/>

¹⁰<https://pillow.readthedocs.io/en/stable/>

dát. Ďalším krokom predspracovania dát bolo konvertovanie farebnej škály snímok na odtiene sivej a zníženie rozlíšenia vstupnej snímky na 256×256 . Na Obrázku 3-2 je v modrom rámečku znázornený proces predspracovania dát pre koronálne diery.



Obrázok 3 – 2 Predspracovanie a augmentácia dát

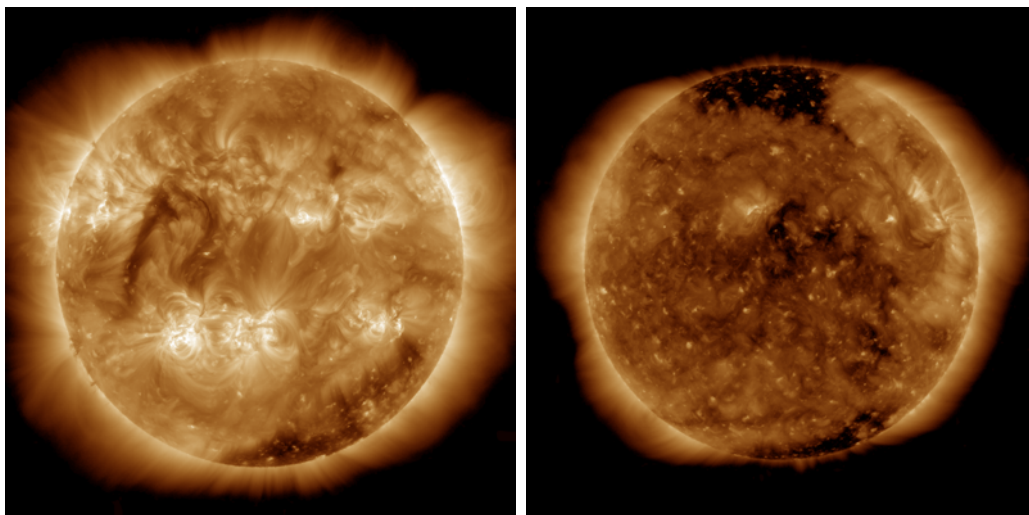
Augmentácia dát

Neuronové siete sú známe tým, že pre dosiahnutie dobrých výsledkov vyžadujú veľké množstvo dát. Z veľkej dátovej množiny dokáže neurónová sieť extrahovať lepšie príznaky aj pri horšej kvalite dát (Perez and Wang, 2017). Avšak, získanie veľkého množstva označených dát vyžaduje veľmi veľa času a ľudských zdrojov, pretože tento proces zvyčajne spočíva v manuálnom označovaní dát. Augmentácia dát je najlepším spôsobom ako vyriešiť tento problém. Aplikáciou malých zmien ako napríklad rotácia alebo prevrátenie snímky, vytvoríme nové príklady, ktoré môžu značne zlepšiť výsledky neurónovej siete (Shorten and Khoshgoftaar, 2019; Mikołajczyk and Grochowski, 2018). Pri augmentácii dát je nutné dbať na to aby použité techniky augmentácie boli relevantné danej oblasti, napríklad pre naše dáta by bolo nevhodné snímky rozťahovať alebo zväčšovať, pretože v našej dátovej množine sa táka snímka nevyskytuje a ani sa tam objaviť nemôže.

Pre zväčšenie dátovej množiny sme rotovali snímky o 90, 180 a 270 stupňov. Týmto spôsobom sme získali štvornásobne väčšiu dátovú množinu. Okrem aplikovania rotácií sme taktiež využili ImageDataAugmentor, ktorý aplikoval nasledujúce transformácie s 50% pravdepodobnosťou:

- Horizontálne prevrátenie – prevrátenie snímky v ose y .
- Vertikálne prevrátenie – prevrátenie snímky v ose x .
- Rotácia – rotácia snímky v rozmedzí $(-45, 45)$ stupňov. Na extrapoláciu bola využitá metóda `cv2.BORDER_REPLICATE`, ktorá zachovala biele resp. čierne pozadie.
- Náhodná gama – náhodný posun gama v rozmedzí $(100, 150)$.
- Náhodný jas a kontrast – náhodná zmena jasnosti v rozmedzí $(0, 0.2)$ a kontrastu v rozmedzí $(0, 0.4)$.

Transformácie vo forme zmeny jasnosti, kontrastu a gamy boli použité na vykompenzovanie snímok slnka, ktoré sa javia byť jasnejšie resp. tmavšie, pozri Obrázok [3–3](#). Takéto jasnejšie snímky sa nachádzajú v rozmedzí rokov 2012-2016 počas maxima slnečnej aktivity, čiže ich máme v značne väčšom počte a niektoré modely, ktoré boli učené na takýchto svetlejších snímkach dosahovali horšie segmentácie na snímkach tmavých.



Obrázok 3–3 Rozdiel v intenzite jasnosti na snímkach z roku 2010 (vľavo) a z roku 2018 (vpravo) – SDO/AIA

Na augmentáciu dát bola použitá Python knižnica `Albumentations` (Buslaev et al. 2020) spolu s generátorom `ImageDataAugmentor` (Tukiainen, 2019). Ukážky týchto augmentácií možno vidieť vo fialovom rámečku na Obrázku 3-2.

Rozdelenie dátových množín

Ako už bolo spomenuté v podkapitole 2.3, k dispozícií máme snímky Slnka z dvoch rôznych filtrov (171Å a 193Å) a niekoľko rôznych zdrojov anotácií, z ktorých sa budú odvíjať naše vstupné dátové množiny, pozri Tabuľky 3-1 a 3-2. Každú z týchto dátových množín použijeme na tréňovanie samostatného modelu, ktorý budeme následne testovať na validačnej množine pozostávajúcej z posledných 10% snímok dátovej množiny a na nezávislej testovacej množine. Žiadna snímka z testovacej a validačnej množiny nebola použitá na tréňovanie modelu.

Testovacie množiny:

- 193Å– 353 snímok z roku 2017 + anotácie Region Growth
- 171Å– 360 snímok z roku 2016 + anotácie SPoCA

Tabuľka 3 – 1 Prehľad získaných anotácií CH

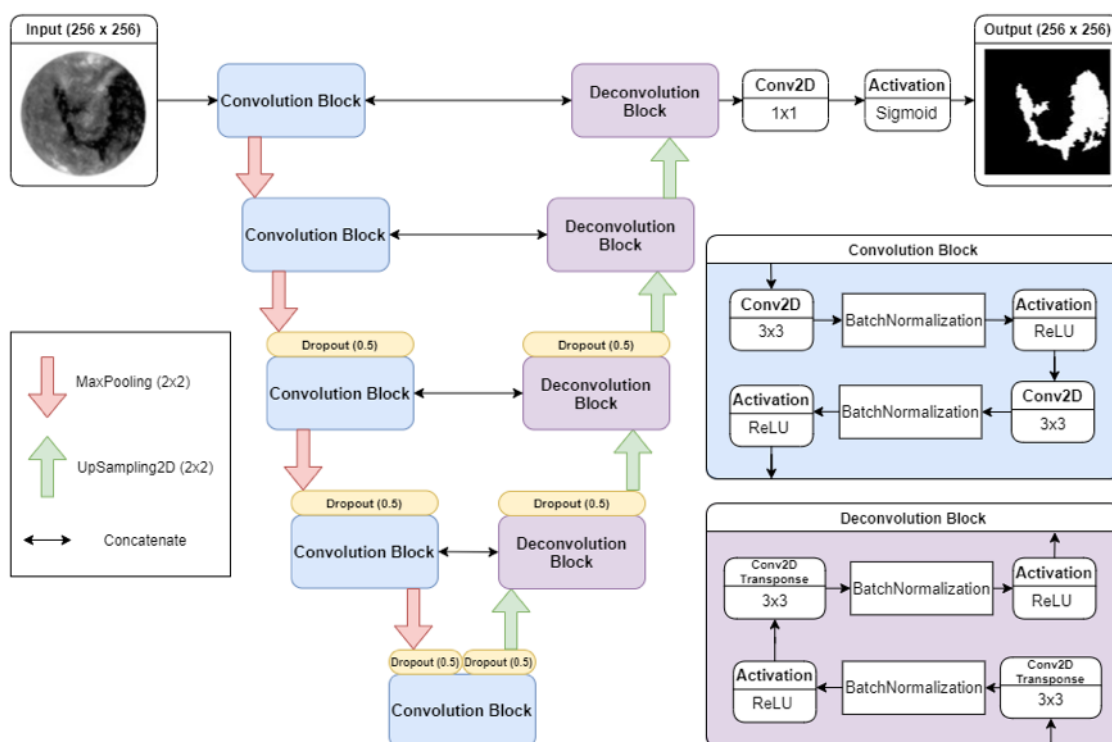
Zdroj	Počiatkové	Rotované	Tréningová množina	Validačná množina
Vlastné	537	2148	1933	215
SPoCA	537	2148	1933	215
Chimera	653	2612	2351	261
Region growth	1968	7872	7085	787

3.3 SCSS-Net

Architektúra nášho modelu (znázornená na obrázku 3–4), ktorú sme nazvali SCSS-Net (Solar Coronal Structures Segmentation Network) je inšpirovaná architektúrou neurónovej siete U-Net (Ronneberger et al. 2015), predstavenej v podkapitole 1.8

Tabuľka 3–2 Prehľad získaných anotácií AR

Zdroj	Počiatkové	Rotované	Trénovacia množina	Validačná množina
Vlastné	533	2132	1919	213
SPoCA	1091	4364	3928	436
SPoCA + vlastné	1607	6428	5785	643



Obrázok 3–4 SCSS-Net architektúra

Vstupom našej siete je snímka Slnka vo veľkosti 256×256 pixelov. Takúto veľkosť sme zvolili pretože pri tomto vstupe boli predikcie dostatočne presné a zároveň

časová a hardvérová náročnosť výpočtov bola relatívne nízka. Rovnako ako bežné konvolučné neuronové siete je SCSS-Net zložený z enkodéra a dekodéra:

- **Enkodér** (ľavá časť) – je zložený z piatich konvolučných blokov. Tieto konvolučné bloky obsahujú dve 3×3 konvolúcie nasledované batch normalizáciou a aktivačnou funkciou ReLU. Pre prvé štyri konvolučné vrstvy platí, že na konci každej z nich nasleduje `max-pooling` 2×2 a zdvojnásobujeme počet konvolučných filtrov (32, 64, 128, 256, 512). Pri poslednej vrstve, resp. konvolučnom bloku sa tento proces otočí, počet filtrov sa znovu zmenší o polovicu, nasleduje 2×2 `upsampling` a začína dekodér.
- **Dekodér** (pravá časť) – pozostáva zo štyroch dekonvolučných blokov, ktoré sa líšia od konvolučných blokov tým, že namiesto konvolúcií využívajú transponované konvolúcie. Pred každým dekonvolučným blokom dochádza k nadzorkovaniu - `upsampling`, a zníženiu filtrov o polovicu.

Na prepojenie snímok a príznakov medzi jednotlivými úrovňami siete využívame funkciu `concatenate` – tá nám umožňuje spojiť pôvodnú vstupnú snímku s naučenými príznakmi. Sieť je regularizovaná pomocou funkcie `dropout` s 50% pravdepodobnosťou počnúc tretím konvolučným blokom a končiac druhým dekonvolučným blokom. Výstup tvorí konvolučná vrstva o veľkosti 1×1 s aktivačnou funkciou `sigmoid`. Na výstupe tak získavame segmentačné masky, ktoré pre každý pixel určujú s akou pravdepodobnosťou daný pixel patrí do danej triedy.

Optimalizácia parametrov SCSS-Net

Aby náš model dosiahol čo najlepšie výsledky, upravovali sme rôzne parametre a hyperparametre modelu. Najprv sme sa zamerali na parametre, ktoré mali dopad na hardvérové nároky modelu – vstupná veľkosť snímky, `batch-size` a počet konvolučných filtrov modelu. Prioritne sme chceli vytvoriť model, ktorý je ľahko reprodukovateľný na bežnom hardvérovom vybavení počítača. Model sme trénovali na

grafickej karte NVIDIA GeForce GTX 1060 Max-Q. Na to aby nám stačila jej 6GB pamäť sme zvolili vstupnú veľkosť snímky 256×256 a `batch-size` o veľkosti 20 snímok. Pri 64 konvolučných filtroch sme dosiahli zlepšenia výsledkov o 1-3%, ale aj pri `batch-size=1` nám model hlásil chybovú hlášku `OutOfMemory`, preto sme nakoniec zvolili 32 konvolučných filtrov. Následne sme využili aj väčšiu výpočtovú silu GPU Tesla K40 s 12GB pamäťou, kde sme robili experimenty na väčších vstupných obrázkoch, väčším počtom konvolčných filtrov a podobne. Došli sme však k záveru, že takéto modely sa často preučili. Preto sme ostali pri štandardnom hardvérovom vybavení a vyššie predstavených nastaveniach modelu.

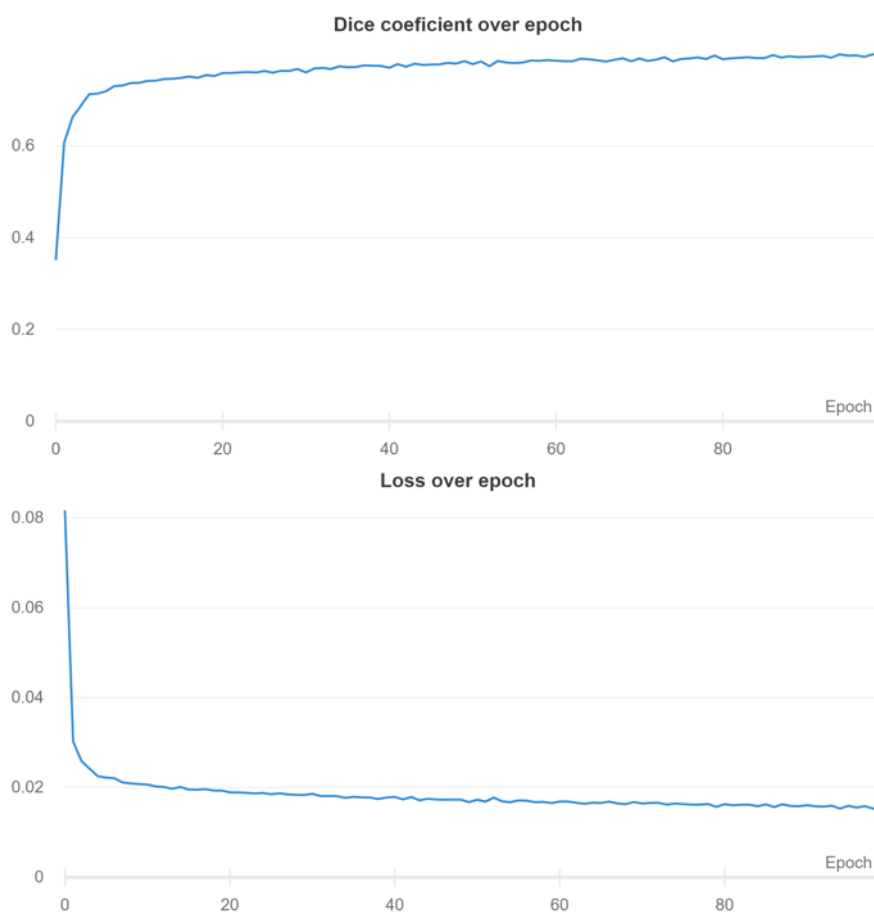
Taktiež sme vyskúšali rôzne optimalizačné funkcie, z ktorých najlepšie výsledky model dosahoval pri použití optimizera Adam a RMSprop. Výsledky modelu trénovaného na rovnakých dátach za použitia rôznych optimalizačných funkcií môžeme vidieť v Tabuľke 3-3. V ďalších experimentoch sme využívali populárny optimizer Adam.

Tabuľka 3-3 Výsledky modelov za použitia rôznych optimalizačných funkcií

Optimalizačná funkcia	Dice	Dice threshold
Adam	0.82	0.87
Adagrad	0.70	0.83
SGD	0.74	0.83
RMSprop	0.82	0.87

Na Obrázku 3-5 sa nachádza priebeh učenia modelu trénovaného na region growth anotáciach. Natrénovať tento model na 7872 snímkach trvalo 7 hodín a 10 minút, pričom jedna epocha trvala približne 4 minúty. Ďalšie grafy priebehu tohto trénovania, zobrazenia systémového a procesného využitia CPU, GPU počas trénovania modelu, grafickú ukážku modelu či samotný konzolový výpis priebehu učenia,

sú dostupné online^[11] a boli vytvorené pomocou API Weights & Biases^[12]. V budúcej práci sa budeme venovať komplexnej vizualizácii všetkých modelov a vizuálnemu porovnaniu priebehov a závislostí jednotlivých metrík pomocou tohto API.



Obrázok 3–5 Priebeh učenia

3.4 Post-processing

Ako už bolo spomenuté, výstupom modelu je v našom prípade pravdepodobnosťou z akou daný pixel snímky je koronálna diera, resp. aktívna oblasť. Nami požadovaný výstup je však:

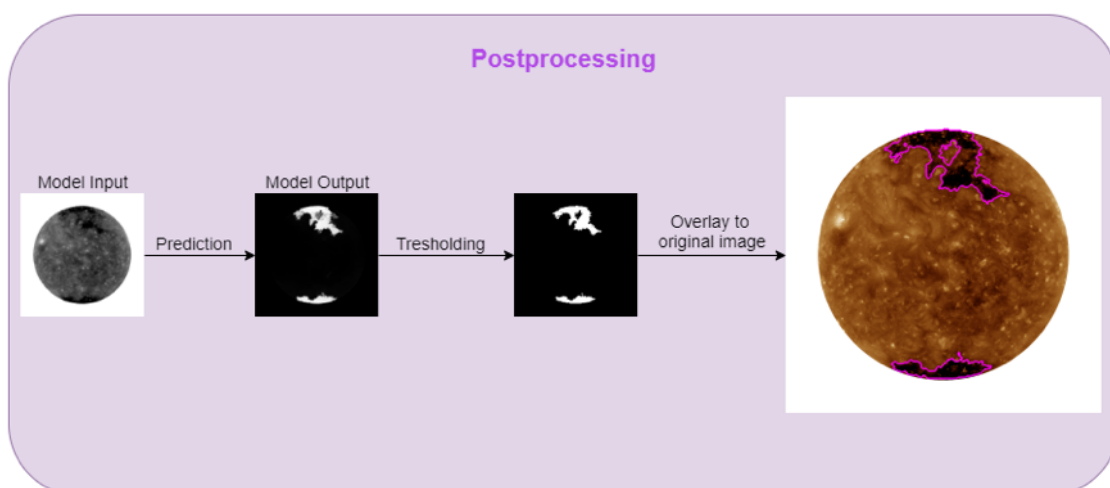
1. binárna maska, ktorá môže slúžiť na štatistickú analýzu snečných štruktúr,

¹¹<https://wandb.ai/matinus/sun-segmentation/runs/2p0ciuvk/>

¹²<https://wandb.ai/site>

- snímka slnka, na ktorej bude vykreslená oblasť, kde sa daná slnečná štruktúra nachádza. Takáto snímka môže slúžiť na prezentáciu a vizuálnu analýzu.

V niekoľkých krokoch, ktoré sú zobrazené na Obrázku 3-2 sme výstup dostatočne spracovali aby získal nami požadovanú podobu. Pre získanie binárnej segmentačnej masky sme všetky hodnoty menšie než 0.5 zmenili na 0 a väčšie na 1. Aplikovaním prahovania sme tak získali čiernobiely snímku, kde biele pixely boli tie, ktoré model označil za koronálnu dieru, resp. aktívnu oblasť. Pre získanie kontúr sme využili metódy počítačového videnia na nájde hrán, tzv. edge detection. Nájdene kontúry boli vo veľkosti vstupu (256×256), ktoré sme následne prepočítali na pôvodnú veľkosť snímky (1024×1024).



Obrázok 3-6 Post-processing

3.5 Vyhodnotenie výsledkov

Model, ktorý sme predstavili v podkapitole 3.3 sme trénovali samostatne pre úlohu segmentácie koronálnych dier na štyroch dátových množinách popísaných v Tabuľke 3-1 a model slúžiaci na segmentáciu aktívnych oblastí bol trénovaný na troch dátových množinách predstavených v Tabuľke 3-2. Natrénovali sme tak 4 modely slúžiace na segmentáciu CH a 3 modely slúžiace na segmentáciu AR. Každý z týchto

modelov bol trénovaný na vlastnej dátovej množine a bol testovaný oproti 10% dát z rovnakého zdroja anotácií a zároveň sme tieto modely testovali na nezávislej dátovej množine. Pre testovanie segmentácií koronálnych dier sme zvolili snímky z roku 2017 a vytvorili sme dve testovacie množiny. Jedná z testovacích množín bola zložená z Region Growth anotácií pretože tieto anotácie sme pokladali za najlepšie a zároveň sme ich mali v najväčšom počte. Druhá testovacia množina bola zložená z anotácií SPoCA. Na testovanie segmentácií aktívnych oblastí sme vybrali snímky roku 2016 a anotácie z algoritmu SPoCA pretože naše vlastné anotácie sa pohybovali v rozmedzí rokov 2012-2015 a nemáme ich k dispozícií vo veľkom počte.

Každý z modelov sme trénovali počas 100 epoch, pričom sme využili funkciu `CallbackCheckpoint`, ktorá ukladá model iba v tej epoche, v ktorej bola chyba najmenšia. Pomocou tejto funkcie môžeme trénovať model počas toľkých epoch a nemusíme sa báť toho, že posledný uložený model nebude ten najlepší.

3.5.1 Koronálne diery

Na Tabuľke [3-4](#) môžeme vidieť, že model dosahoval dostatočne dobre výsledky na validačných množinách aj v prípade modelu, ktorý bol učný na našich vlastných anotáciách, ktoré sme nemali k dispozícií vo veľkom počte. Naopak na výsledkoch modelu trénovaného na dátovej množine zloženej zo SPoCA anotácií, ktoré boli v rovnakom počte ako naše vlastné anotácie, sa nám potvrdzuje, že tieto anotácie nie sú dokonalé a obsahujú veľa chýb. V prípade modelov, trénovaných na dátovej množine CHIMERA a Region Growth vidíme, že na validačnej množine dosiahli podobné výsledky a z toho môžeme usúdiť, že výrazne rozdielna veľkosť dátovej množiny (2351 - CHIMERA a 7085 - Region Growth) nemusela mať veľký vplyv.

Výsledky z validačnej množiny sa však nehodia na celkové vyhodnotenie modelov a ich porovnanie medzi sebou. Na rozhodnutie, ktorý z natrénovaných modelov je najlepší sme tieto modely otestovali na dvoch nezávislých testovacích množinách. V Tabuľke [3-5](#) môžeme vidieť výsledky na testovacej množine zloženej z anotácií SPoCA. Najlepšie výsledky dosiahol model trénovaný na region growth anotáciách.

Tabuľka 3 – 4 Výsledky modelov (CH) na validačnej množine

Trénovacia množina	Validačná množina	Dice	IoU
Vlastné	215	0.85	0.74
SPoCA	215	0.74	0.59
Chimera	261	0.86	0.76
Region growth	787	0.88	0.78

Modely trénované na SPoCA a CHIMERA anotáciách dosiahli porovnateľné výsledky, avšak aj napriek tomu, že žiadne snímky z testovacej množiny neboli použité na tréovanie, tak v prípade SPoCA modelu to mohlo ovplyvniť výsledky. Na tejto testovacej množine si mal najhoršie výsledky model trénovaný na našich vlastných anotáciách.

Tabuľka 3 – 5 Výsledky modelov na testovacej množine zloženej zo SPoCA anotácií

Trénovacia množina	Testovacia množina	Dice	IoU
Vlastné	353	0.56	0.39
SPoCA	353	0.59	0.42
Chimera	353	0.59	0.42
Region growth	353	0.63	0.46

Avšak ak sa pozrieme na Tabuľku 3–6, ktorá obsahuje výsledky jednotlivých modelov testovaných na region growth anotáciách, tak môžeme vidieť, že model SPoCA mal oveľa horšie výsledky. Naopak zvyšné modely majú oveľa lepšie výsledky, z ktorých najlepší je model Region growth za ním CHIMERA a model trénovaný na vlastných anotáciách.

Na Obrázku 3–7, kde veľkými písmenami sú označené snímky s pôvodnými anotáciami (A - vlastné, B - SPoCA, C - Region growth, D - CHIMERA) a malými písmenami snímky segmentované modelom (a - vlastné, b - SPoCA, c - Region gro-

Tabuľka 3–6 Výsledky modelov na testovacej množine zloženej z Region Growth anotácií

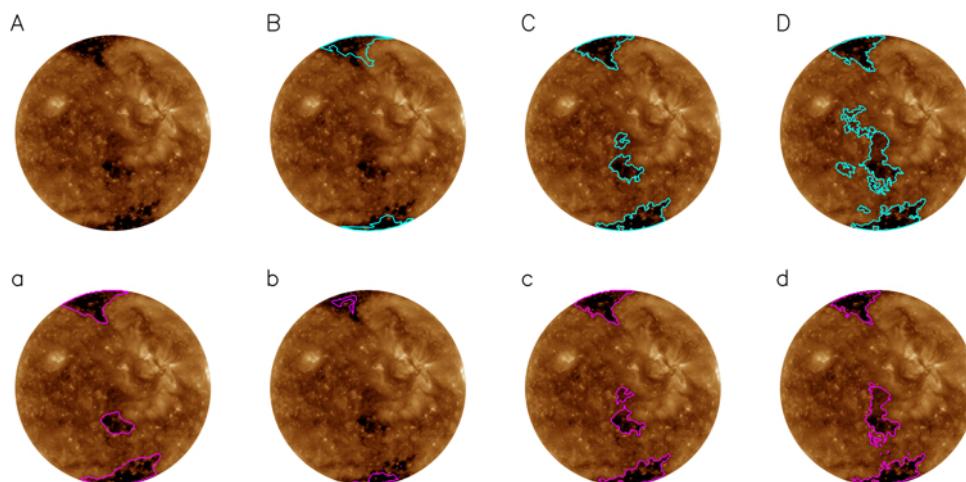
Trénovacia množina	Testovacia množina	Dice	IoU
Vlastné	353	0.83	0.71
SPoCA	353	0.35	0.21
Chimera	353	0.85	0.73
Region growth	353	0.88	0.78

wth, d - CHIMERA) môžeme spozorovať, že výsledná DICE metrika SPoCA modelu na SPoCA testovacej množine bola naozaj ovplyvnená rovnakým zdrojom anotácií. Napriek tomu, že SPoCA anotácie nie sú práve najkvalitnejšie, tak sa SPoCA model odnaučil segmentovať tie časti snímky, ktoré už nie sú koronálna diera. Podobný jav môžeme spozorovať aj pri modeli CHIMERA, kde pôvodne anotácie (D) preceňujú veľkosť koronálnych dier, ale model už tieto oblasti, ktoré sa za koronálne diery už nepovažujú, neoznačuje ako CH. Model trénovaný na našich vlastných anotáciách (A^{13}) sa dokázal naučiť veľmi presne segmentovať koronálne diery aj napriek tomu, že sme mali k dispozícii iba limitovaný počet anotácií. Vďaka veľkému počtu kvalitných anotácií sú segmentácie modelu Region growth takmer rovnaké ako pôvodné anotácie. Ukážku segmentácií vytvorených modelom, ktorý bol trénovaný na region growth anotáciách, môžeme vidieť na Obrázku 3–8.

3.5.2 Aktívne oblasti

Segmentácia aktívnych oblastí bola zložitejšia, pretože sme nemali k dispozícii dostatočné množstvo kvalitných anotácií. SPoCA anotácie obsahovali veľké množstvo nesprávne označených oblastí a nakoľko ručná anotácia snímok je veľmi časovo náročná, tak vlastných anotácií máme v príliš malom množstve a preto sme sa rozhodli vytvoriť testovaciu dátovú množinu iba zo SPoCA anotácií. Na základe týchto faktov sme sa pri segmentácií aktívnych oblastí spoliehali hlavne na vizuálne porovnanie

¹³Vlastnú anotáciu sme nemali k dispozícii pre túto snímku.



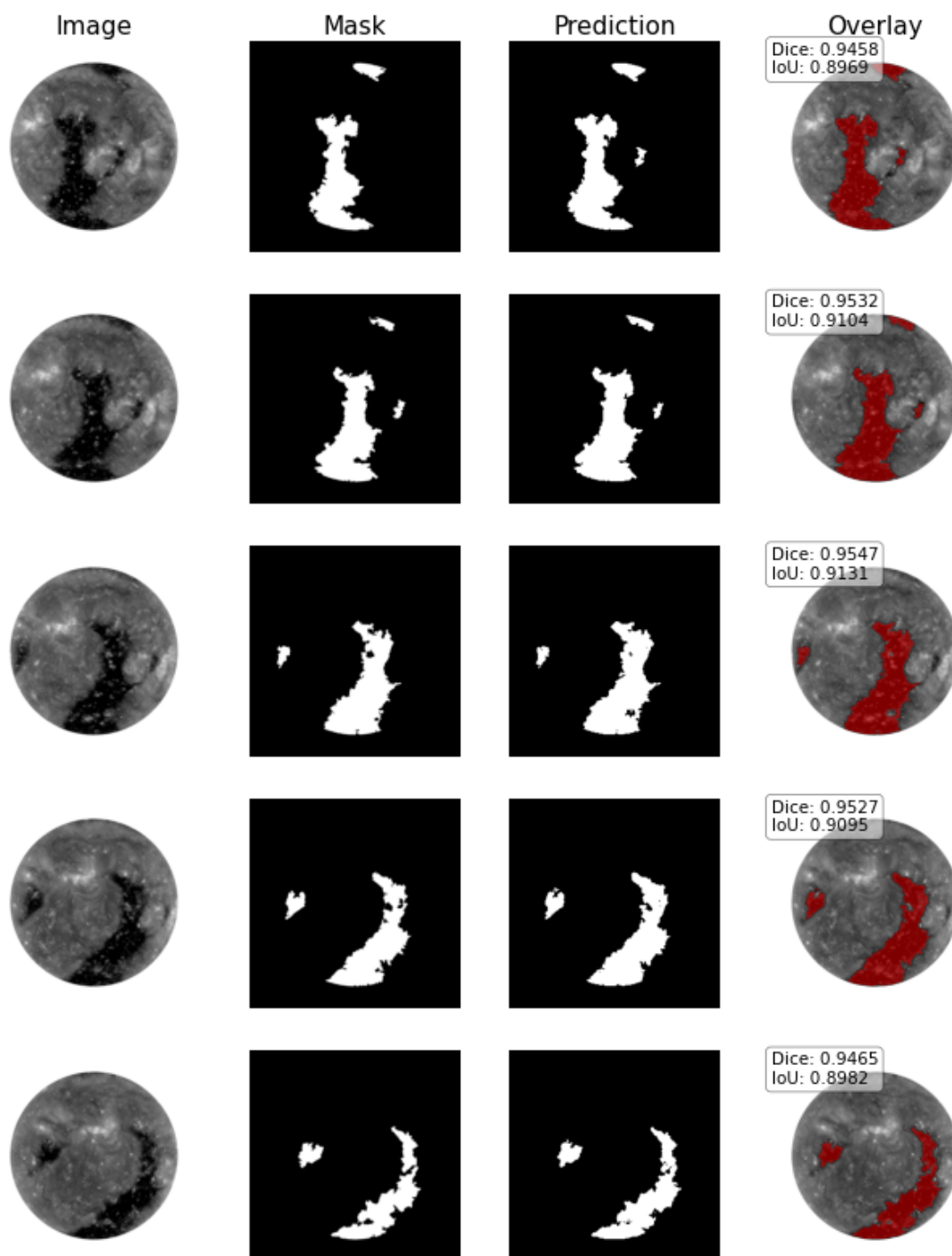
Obrázok 3 – 7 Porovnanie segmentácií koronálnych dier

Tabuľka 3 – 7 Výsledky modelov (AR) na validačnej množine

Trénovacia množina	Validačná množina	Dice	IoU
Vlastné	213	0.81	0.68
SPoCA	436	0.71	0.55
SPoCA + vlastné	643	0.51	0.34

segmentácií. Na Tabuľke 3–7 model SPoCA na segmentáciu AR dosiahol porovnateľne výsledky na validačnej množine ako model SPoCA na segmentáciu CH. Na validačnej množine dosiahol najlepšie výsledky model učný na našich vlastných anotáciách a najhoršie ten, ktorý bol učný na oboch dátových množinách. Ukážku segmentácií vytvorených modelom, ktorý bol trévaný na spojenej dátovej množine, môžeme vidieť na Obrázku 3–10.

Ak sa pozrieme na výsledky z testovacej množiny v Tabuľke 3–8 tak to pre model trévaný na našich vlastných anotáciách vyzerá horšie a výsledok modelu trévaného na spojenej množine je približný priemer zvyšných modelov. Preto aby sme mohli lepšie model vyhodnotiť, tak sa musíme pozrieť na výsledne anotácia a rozhodnúť sa na základe nich.



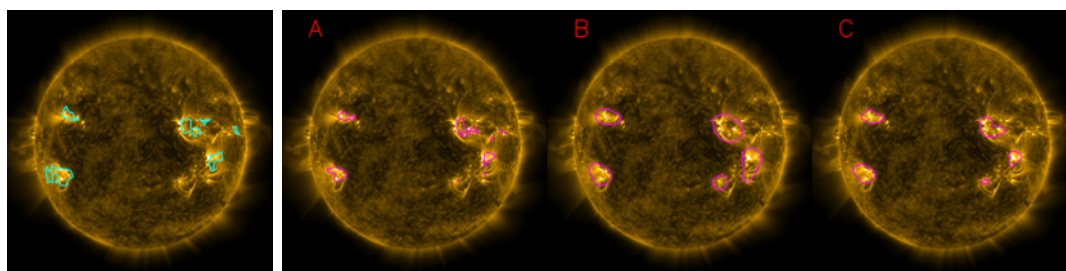
Obrázok 3–8 Ukážka segmentácií koronálnych dier

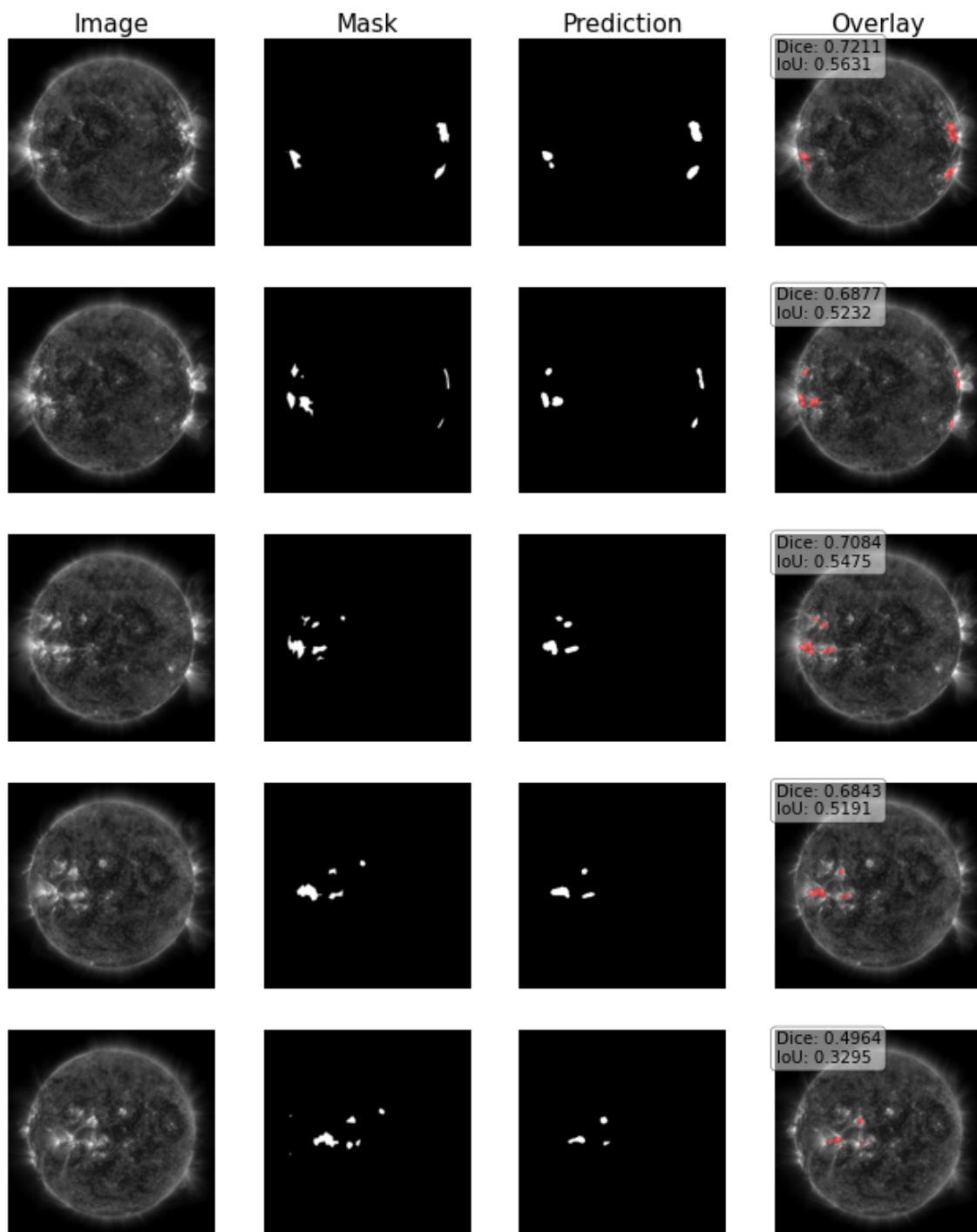
Na Obrázku 3–9 sa nachádza pôvodná SPoCA anotácia (vľavo) a písmenami označené sú označené segmentácie jednotlivých modelov (A - SPoCA, B - vlastné, C -

Tabuľka 3 – 8 Výsledky modelov na testovacej množine

Trénovacia množina	Testovacia množina	Dice	IoU
Vlastné	360	0.44	0.28
SPoCA	360	0.68	0.51
SPoCA + vlastné	360	0.56	0.39

SPoCA + vlastné). SPoCA model segmentoval menšie oblasti, ale zato presnejšie než sú pôvodné anotácie zatiaľ čo, model naučený na našich anotáciách (B) segmentoval AR dôkladnejšie resp. segmentoval väčšie oblasti. Ak sa pozrieme na Obrázok 3–9, tak môžeme vidieť, že model sa naučil aj „štýl“ anotácie a je vidieť, ktorý model bol trébovaný na anotáciách z algoritmu, a ktorý na anotáciách vytvorených človekom. Po bližšom posúdení výsledných snímok sme sa rozhodli, že model naučený na spojenej dátovej množine (C) segmentuje aktívne oblasti najpresnejšie. Tento model segmentuje AR dostatočne presne a zároveň nevytvára až také oblúkovité oblasti, ktoré segmentujú aj časti slnka, ktoré už AR nie sú.

**Obrázok 3 – 9** Porovnanie segmentácií aktívnych oblastí



Obrázok 3 – 10 Ukážka segmentácií aktívnych oblastí

4 Záver

Cieľom tejto práce bolo vytvoriť model hlbokoj neurónovej siete určenej na segmentáciu koronálnych dier a aktívnych oblastí. V prvej časti práce sme sa venovali teoretickému prehľadu o využitých metódach a o samotnom hlbokom učení. Bližšie sme sa zamerali na konvolučné neuronové siete a ich architektúry využívané na segmentačné úlohy.

Ďalej sme sa pozreli na kozmické počasie, na jeho pôvod a možné dopady na život na Zemi. Zamerali sme sa na pôvodcu vesmírneho počasia, ktorým je Slnko, a na javy odohrávajúce sa v jeho atmosfére. Bližšie sme opísali dve solárne štruktúry, ktorými boli koronálne diery a aktívne oblasti, pričom tie mal za úlohu segmentovať náš model neurónovej siete. Pozreli sme sa bližšie na súčasne metódy detekcie slnečných štruktúr a na práce venujúce sa tejto problematike. Z týchto prác bola pre nás motiváciou práve práca [Illarionov and Tlatov \(2018\)](#), pričom sme sa snažili o prekonanie ich výsledkov, čo sa nám aj podarilo.

Počas vypracovávania tejto práce sa nám podarilo nájsť a zozbierať dáta koronálnych dier z niekoľkých zdrojov, vďaka čomu sme mohli preskúmať dáta o rôznej kvalite a počte a ich dopadu na výsledný segmentačný model. Zistili sme, že ak máme dáta dobrej kvality, napríklad anotácie vytvorené manuálne nie je ich potrebné až také veľké množstvo. Naopak ak máme dáta v horšej kvalite, ktorými boli v našom prípade anotácie SPoCA, tak ich potrebujeme veľmi veľa. Najlepší prípad však je ak máme dostatok dát vo veľmi dobrej kvalite, ako v prípade region growth anotácií. Vďaka týmto anotáciám, ktoré boli využité aj v práci [Illarionov and Tlatov \(2018\)](#) sa nám podarilo prekonať ich výsledky o približne o 6%.

V prípade modelu trénovaného za účelom segmentácie aktívnych oblastí sme dosiahli najpresnejšie výsledky po zväčšení dátovej množiny spojením kvalitnejších manuálnych anotácií s horšími, ale zato početnejšími SPoCA anotáciami. Keďže sme nemali žiadne kvalitné anotácie aktívnych oblastí, ktoré by sme mohli využiť ako referenčné anotácie, pri vyhodnocovaní tohto modelu zamerali hlavne na vizuálne

vyhodnotenie modelu.

Segmentačné modely môžeme vyhodnotiť ako úspešne, avšak vždy sa nájde priestor na zlepšenie alebo rozšírenie. Jedným z takýchto rozšírení môže byť vytvorenie jednotného modelu, ktorý by segmentoval aktívne oblasti, ale aj koronálne diery za použitia dvoch vstupných snímok v podobe 171Å a 193Å. Táto práca vznikla v spolupráci s RNDr. Šimonom Mackovjakom, PhD., ktorý pracuje na Oddelení kozmickej fyziky ÚEF, SAV a spolu s ním, vedúcim tejto práce doc. Ing. Petrom Butkom, PhD. a konzultantkou Ing. Vierkou Maslej Krešňákovou sme pripravili odborný článok s cieľom jeho publikácie v karentovanom časopise Monthly Notices of the Royal Astronomical Society (skr. MNRAS).

Literatúra

Active Regions on the Sun (2021). <https://scied.ucar.edu/sun-active-region>.

Aghdam, H. H. and Heravi, E. J. (2017). Guide to convolutional neural networks, *New York, NY: Springer* **10**: 978–973.

Ballard, D. H., Hinton, G. E. and Sejnowski, T. J. (1983). Parallel visual computation, *Nature* **306**(5938): 21–26.

Burrell, A. G., Halford, A., Klenzing, J., Stoneback, R. A., Morley, S. K., Annex, A. M., Laundal, K. M., Kellerman, A. C., Stansby, D. and Ma, J. (2018). Snakes on a spaceship—an overview of python in heliophysics, *Journal of Geophysical Research: Space Physics* **123**(12): 10,384–10,402.

Buslaev, A., Iglovikov, V. I., Khvedchenya, E., Parinov, A., Druzhinin, M. and Kalinin, A. A. (2020). Albumentations: Fast and flexible image augmentations, *Information* **11**(2).

Camporeale, E. (2019). The challenge of machine learning in space weather: Nowcasting and forecasting, *Space Weather* **17**(8): 1166–1207.

Canny, J. (1986). A computational approach to edge detection, *IEEE Transactions on pattern analysis and machine intelligence* (6): 679–698.

Coronal Holes (2021). <https://www.swpc.noaa.gov/phenomena/coronal-holes>.

De Boer, P.-T., Kroese, D. P., Mannor, S. and Rubinstein, R. Y. (2005). A tutorial on the cross-entropy method, *Annals of operations research* **134**(1): 19–67.

Dechter, R. (1986). Learning while searching in constraint-satisfaction-problems, *Proceedings of the Fifth AAAI National Conference on Artificial Intelligence, AAAI'86*, AAAI Press, p. 178–183.

-
- Delouille, V., Hofmeister, S. J., Reiss, M. A., Mampaey, B., Temmer, M. and Veronig, A. (2018). Chapter 15 - coronal holes detection using supervised classification, *in* E. Camporeale, S. Wing and J. R. Johnson (eds), *Machine Learning Techniques for Space Weather*, Elsevier, pp. 365–395.
- Delouille, V., Mampaey, B., Verbeeck, C. and de Visscher, R. (2012). The spoc-suite: a software for extraction and tracking of active regions and coronal holes on euv images.
- Duchi, J., Hazan, E. and Singer, Y. (2011). Adaptive subgradient methods for online learning and stochastic optimization., *Journal of machine learning research* **12**(7).
- Garner, R. (2015). Solar storm and space weather - frequently asked questions. https://www.nasa.gov/mission_pages/sunearth/spaceweather/index.htm
- Garton, Tadhg M., Gallagher, Peter T. and Murray, Sophie A. (2018). Automated coronal hole identification via multi-thermal intensity segmentation, *J. Space Weather Space Clim.* **8**: A02.
- Glorot, X., Bordes, A. and Bengio, Y. (2011). Deep sparse rectifier neural networks, *in* G. Gordon, D. Dunson and M. Dudík (eds), *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, Vol. 15 of *Proceedings of Machine Learning Research*, JMLR Workshop and Conference Proceedings, Fort Lauderdale, FL, USA, pp. 315–323.
- Goodfellow, I., Bengio, Y. and Courville, A. (2016). *Deep Learning*, MIT Press. <http://www.deeplearningbook.org>
- Han, J. and Moraga, C. (1995). The influence of the sigmoid function parameters on the speed of backpropagation learning, *in* J. Mira and F. Sandoval (eds), *From Natural to Artificial Neural Computation*, Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 195–201.
-

-
- Hartigan, J. A. and Wong, M. A. (1979). Algorithm as 136: A k-means clustering algorithm, *Journal of the royal statistical society. series c (applied statistics)* **28**(1): 100–108.
- He, K., Zhang, X., Ren, S. and Sun, J. (2015). Delving deep into rectifiers: Surpassing human-level performance on imagenet classification, *CoRR* **abs/1502.01852**.
- Henney, C. and Harvey, J. (2005). Automated coronal hole detection using he i 1083 nm spectroheliograms and photospheric magnetograms, *arXiv: Astrophysics* .
- Hurlburt, N., Cheung, M., Schrijver, C., Chang, L., Freeland, S., Green, S., Heck, C., Jaffey, A., Kobashi, A., Schiff, D., Serafin, J., Seguin, R., Slater, G., Somani, A. and Timmons, R. (2012). Heliophysics Event Knowledgebase for the Solar Dynamics Observatory (SDO) and Beyond, **275**(1-2): 67–78.
- Illarionov, E. A. and Tlatov, A. G. (2018). Segmentation of coronal holes in solar disc images with a convolutional neural network, **481**(4): 5014–5021.
- Jatla, V., Pattichis, M. S. and Arge, C. N. (2020). Image processing methods for coronal hole segmentation, matching, and map classification, *IEEE Transactions on Image Processing* **29**: 1641–1653.
- Kiefer, J., Wolfowitz, J. et al. (1952). Stochastic estimation of the maximum of a regression function, *The Annals of Mathematical Statistics* **23**(3): 462–466.
- Kingma, D. P. and Ba, J. (2017). Adam: A method for stochastic optimization.
- Krista, L. D. and Gallagher, P. T. (2009). Automated coronal hole detection using local intensity thresholding techniques, *Solar Physics* **256**(1-2): 87–100.
- Krizhevsky, A., Sutskever, I. and Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks, in F. Pereira, C. J. C. Burges, L. Bottou and K. Q. Weinberger (eds), *Advances in Neural Information Processing Systems*, Vol. 25, Curran Associates, Inc.
-

-
- Lecun, Y., Bengio, Y. and Hinton, G. (2015). Deep learning, **521**(7553): 436–444.
- Legendre, P. and Legendre, L. (2012). *Numerical ecology*, Elsevier.
- Long, J., Shelhamer, E. and Darrell, T. (2014). Fully convolutional networks for semantic segmentation, *CoRR* **abs/1411.4038**.
- Maas, A. L., Hannun, A. Y. and Ng, A. Y. (2013). Rectifier nonlinearities improve neural network acoustic models, *Proc. icml*, Vol. 30, Citeseer, p. 3.
- McCulloch, W. S. and Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity, *The bulletin of mathematical biophysics* **5**(4): 115–133.
- Mikołajczyk, A. and Grochowski, M. (2018). Data augmentation for improving deep learning in image classification problem, *2018 international interdisciplinary PhD workshop (IIPhDW)*, IEEE, pp. 117–122.
- Murphy, K. P. (2012). *Machine learning: a probabilistic perspective*, MIT press.
- Nielsen, M. A. (2015). *Neural networks and deep learning*, Vol. 25, Determination press San Francisco, CA.
- Paralic, J. (2003). Objavovanie znalostí v databázach.
- Perez, L. and Wang, J. (2017). The effectiveness of data augmentation in image classification using deep learning, *arXiv preprint arXiv:1712.04621* .
- Pesnell, W. D., Thompson, B. J. and Chamberlin, P. (2011). The solar dynamics observatory (sdo), *The Solar Dynamics Observatory*, Springer, pp. 3–15.
- Price-Whelan, A. M., Sipőcz, B. M., Günther, H. M., Lim, P. L., Crawford, S. M., Conseil, S., Shupe, D. L., Craig, M. W., Dencheva, N. and et al. (2018). The astropy project: Building an open-science project and status of the v2.0 core package, *The Astronomical Journal* **156**(3): 123.
-

-
- Reed, R. D. and Marks, R. J. (1998). *Neural Smithing: Supervised Learning in Feedforward Artificial Neural Networks*, MIT Press, Cambridge, MA, USA.
- Rios, J., Alanis, A., Arana-Daniel, N., Lopez-Franco, C. and Sanchez, E. (2020). *Neural Networks Modeling and Control: Applications for Unknown Nonlinear Delayed Systems in Discrete Time*, Elsevier Science.
- Robbins, H. (2007). A stochastic approximation method, *Annals of Mathematical Statistics* **22**: 400–407.
- Robitaille, T. P., Tollerud, E. J., Greenfield, P., Droettboom, M., Bray, E., Aldcroft, T., Davis, M., Ginsburg, A., Price-Whelan, A. M. and et al. (2013). Astropy: A community python package for astronomy, *Astronomy Astrophysics* **558**: A33.
- Ronneberger, O., Fischer, P. and Brox, T. (2015). U-net: Convolutional networks for biomedical image segmentation, *International Conference on Medical image computing and computer-assisted intervention*, Springer, pp. 234–241.
- Rumelhart, D. E., Hinton, G. E. and Williams, R. J. (1985). Learning internal representations by error propagation, *Technical report*, California Univ San Diego La Jolla Inst for Cognitive Science.
- Rušin, V. and Rybanský, M. (1990). *Slnečná koróna*, Veda.
- Sahoo, P. K., Soltani, S. and Wong, A. K. (1988). A survey of thresholding techniques, *Computer vision, graphics, and image processing* **41**(2): 233–260.
- Schmidhuber, J. (2015). Deep Learning, *Scholarpedia* **10**(11): 32832.
- Schrijver, C. and Siscoe, G. (2010). *Heliophysics: Evolving Solar Activity and the Climates of Space and Earth*, Cambridge University Press.
- Schuh, M., Angryk, R. and Martens, P. (2016). A large-scale dataset of solar event reports from automated feature recognition modules, *Journal of Space Weather and Space Climate* **6**: A22.

-
- Shorten, C. and Khoshgoftaar, T. M. (2019). A survey on image data augmentation for deep learning, *Journal of Big Data* **6**(1): 60.
- Simpson, R., Page, K. R. and De Roure, D. (2014). Zooniverse: Observing the world’s largest citizen science platform, *Proceedings of the 23rd International Conference on World Wide Web, WWW ’14 Companion*, Association for Computing Machinery, New York, NY, USA, p. 1049–1054.
- Sorensen, T. A. (1948). A method of establishing groups of equal amplitude in plant sociology based on similarity of species content and its application to analyses of the vegetation on danish commons, *Biol. Skar.* **5**: 1–34.
- Space Weather Prediction Center (2021). <https://www.swpc.noaa.gov/>.
- Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I. and Salakhutdinov, R. (2014). Dropout: A simple way to prevent neural networks from overfitting, *Journal of Machine Learning Research* **15**(56): 1929–1958.
- Svozil, D., Kvasnicka, V. and Pospichal, J. (1997). Introduction to multi-layer feed-forward neural networks, *Chemometrics and Intelligent Laboratory Systems* **39**(1): 43–62.
- Taubman, D. S. and Marcellin, M. W. (2002). Jpeg2000: standard for interactive imaging, *Proceedings of the IEEE* **90**(8): 1336–1357.
- The SunPy Community and Barnes, e. a. (2020). The sunpy project: Open source development and status of the version 1.0 core package, *The Astrophysical Journal* **890**: 68–.
- Tieleman, T. and Hinton, G. (2012). Lecture 6.5—RmsProp: Divide the gradient by a running average of its recent magnitude, COURSERA: Neural Networks for Machine Learning.

Tlatov, A., Tavastsherna, K. and Vasil'eva, V. (2014). Coronal holes in solar cycles 21 to 23, *Solar Physics* **289**.

Tukiainen, M. (2019). Imagedataaugmentor, <https://github.com/mjkvaak/ImageDataAugmentor>

Urbán, F. (2020). Získanie anotácií v rámci zvolenej úlohy analýzy astronomických dát.

van Driel-Gesztelyi, L. and Green, L. M. (2015). Evolution of active regions, *Living Reviews in Solar Physics* **12**(1): 1–98.

Šarūnas Raudys (1998). Evolution and generalization of a single neurone: I. single-layer perceptron as seven statistical classifiers, *Neural Networks* **11**(2): 283–296.

Zoznam príloh

Príloha A CD médium – záverečná práca v elektronickej podobe, príručky v elektronickej podobe a zdrojový kód.

Príloha B Používateľská príručka

Príloha C Systémová príručka