Technická univerzita v Košiciach

Fakulta elektrotechniky a informatiky

Využitie strojového učenia pre filtrovanie dát z kozmickej fyziky

Bakalárska práca

2025 **Lukáš Kostár**

Technická univerzita v Košiciach

Fakulta elektrotechniky a informatiky

Využitie strojového učenia pre filtrovanie dát z kozmickej fyziky

Bakalárska práca

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

Košice 2025 **Lukáš Kostár**

Abstrakt v SJ

Dáta získané pozorovaním nočnej oblohy majú využitie v rôznych výskumných oblastiach. Pre veľké objemy obrazových dát, ako sú snímky nočnej oblohy, je ale na ich spracovanie nevyhnutné aplikovať efektívne metódy. Predkladaná bakalárska práca sa zameriava na analýzu snímok nočnej oblohy z celooblohových kamier (All-Sky Immagers ASI) s využitím metód hlbokého učenia. Tieto metódy boli použité na automatickú klasifikáciu do tried podľa stupňa oblačnosti. V praktickej časti práce boli natrénované a otestované konvolučné neurónové siete na anotovaných dátach. Najúspešnejší model bol napokon otestovaný aj na dátach z ďalšej kamery, zo siete prevádzkovanej Bostonskou Univerzitou, s cieľom overiť jeho využiteľnosť v praxi.

Kľúčové slová

hlobké učenie, konvolučné neurónové siete, celooblohová kamera, obrazové dáta, klasifikácia

Abstrakt v AJ

Data obtained by observing the night sky are used in various research areas. However, for large volumes of image data, such as night sky images, it is necessary to apply efficient methods for their processing. The presented bachelor's thesis focuses on the analysis of night sky images from all-sky cameras (All-Sky Imagers ASI) using deep learning methods. These methods were used for automatic classification into classes according to the degree of cloudiness. In the practical part of the work, convolutional neural networks were trained and tested on annotated data. Finally, the most successful model was also tested on data from another camera, from the network operated by Boston University, in order to verify its usability in practice.

Kľúčové slová v AJ

deep learning, convolutional neural networks, all-sky camera, image data, classification

77016

TECHNICKÁ UNIVERZITA V KOŠICIACH

FAKULTA ELEKTROTECHNIKY A INFORMATIKY

Katedra kybernetiky a umelej inteligencie

ZADANIE BAKALÁRSKEJ PRÁCE

Študijný odbor: **Informatika**

Študijný program: **Hospodárska informatika**

Názov práce:

Využitie strojového učenia pre filtrovanie dát z kozmickej fyziky

Using machine learning to filter data from space physics

Študent: **Lukáš Kostár**

Školiteľ: **doc. Ing. Peter Butka, PhD.**

Školiace pracovisko: **Katedra kybernetiky a umelej inteligencie**

Konzultant práce: **RNDr. Šimon Mackovjak, PhD.**

Pracovisko konzultanta: **Ústav experimentálnej fyziky SAV**

Pokyny na vypracovanie bakalárskej práce:

1. Podať teoretický prehľad problematiky metód hlbokého učenia, s dôrazom na spracovanie obrazových dát z oblasti pozorovania svetelného žiarenia atmosféry (tzv. airglow).
2. Pripraviť dáta pre využitie hlbokého učenia pri filtrovaní pozorovaní staníc so zameraním na sledovanie svetelného žiarenia atmosféry.
3. Navrhnúť, realizovať a vyhodnotiť experimenty pre filtrovanie pozorovaní svetelného žiarenia atmosféry.
4. Vypracovať dokumentáciu podľa pokynov katedry a vedúceho práce.

Jazyk, v ktorom sa práca vypracuje: slovenský

Termín pre odovzdanie práce: 23.05.2025

Dátum zadania bakalárskej práce: 31.10.2024

TECHNICKÁ UNIVERZITA V KOŠICIACH FAKULTA ELEKTROTECHNIKY A INFORMATIKY -10-

v.z.

prof. Ing. Liberios Vokorokos, PhD.

dekan fakulty

Čestné vyhlásenie

Vyhlasujem, že som bakalársku prácu vypracoval samostatne s použitím uvedenej odbornej literatúry, ako aj s použitím generatívnych nástrojov umelej inteligencie v súlade s metodickými pokynmi TUKE. Umelá inteligencia bola použitá ako podporný nástroj bez porušenia zásad akademickej etiky a autorskej integrity.

Košice 23. 5. 2025 .....

Vlastnoručný podpis

Poďakovanie

Týmto by som chcel poďakovať vedúcemu mojej bakalárskej práce, doc. Ing. Petrovi Butkovi, PhD., konzultantovi RNDr. Šimonovi Mackovjakovi, PhD. a Ing. Viere Krešňákovej, PhD. za usmernenia, odbornú pomoc a podporu pri vypracovaní tejto bakalárskej práce.

Obsah

Úvod **1**

1 Hlboké učenie **3**

1.1 Úvod do hlbokého učenia 3

1.2 Neurónové siete 3

1.2.1 Aktivačné funkcie 5

1.2.2 Chybové funkcie 7

1.2.3 Optimalizácia 8

1.2.4 Regularizácia 10

1.3 Konvolučné neurónové siete 11

2 Airglow a jeho pozorovanie **15**

2.1 Airglow 15

2.2 Celooblohová kamera 15

2.3 Analýza súčasného stavu 17

3 Využitie strojového učenia pre filtrovanie dát z kozmickej fyziky **22**

3.1 Pochopenie cieľa 22

3.2 Pochopenie dát 24

3.3 Príprava dát 27

3.4 Modelovanie 29

3.4.1 Referenčné modely 29

3.4.2 Konvolučné neurónové siete 31

3.5 Vyhodnotenie 42

3.6 Nasadenie 45

4 Záver **48**

Zoznam príloh **51**

Zoznam obrázkov

1 – 1 Architektúra perceptrónu. 3
1 – 2 Architektúra neurónovej siete. 4
1 – 3 Priebeh funkcie ReLU. 6
1 – 4 Priebeh funkcie Sigmoid. 6
1 – 5 Znázornenie funkcie softmax. 7
1 – 6 Algoritmus spätného šírenie chýb. 9
1 – 7 Podučenie, preučenie a optimálny model. 10
1 – 8 Znázornenie regularizácie pomocou metódy dropout. 11
1 – 9 Jednoduchá konvolučná neurónová sieť (O'shea and Nash, 2015). . 12
1 – 10 Proces konvolúcie. 13
1 – 11 Maxpooling vrstva. 14
1 – 12 Flatten pre jeden a viacero kernelov. 14
2 – 1 Polárna žiara a airglow pozorovaná z Medzinárodnej vesmírnej stanice ISS. Zdroj: NASA. 16
2 – 2 Rozmiestnenie staníc celooblohových kamier, ktoré prevádzkuje Bostnoská Univerzita. Zdroj: `https://sirius.bu.edu/aeronomy/`. 17
2 – 3 Plazmová bublina na snímke zo stanie EL Leoncito 19. decembra 2023. 18
2 – 4 Vizualizácia rozdelenia snímky z celooblohovej kamery v práci Mommert (2020). 20
3 – 1 Matica zámen pre viactriednu klasifikáciu. 23
3 – 2 Jasná nočná obloha (trieda **clear**). 25
3 – 3 Zamračená nočná obloha (trieda **cloudy**). 26
3 – 4 Čiastočne zamračená nočná obloha (trieda **partly**). 26
3 – 5 Matica zámen pre vyhodnotenie klasifikácie rozhodovacieho stromu. 30
3 – 6 Matica zámen pre vyhodnotenie klasifikácie modelu Random Forest. 31
3 – 7 Architektúra prvého jednoduchého modelu A32,64. 32

3 – 8 Matica zámen pre vyhodnotenie klasifikácie modelu A32,64 34
3 – 9 Architektúra druhého modelu B32,64,128. 34
3 – 10 Príklad nesprávnej klasifikácie obrázku modelom B32,64,128. . . . 35
3 – 11 Matica zámen pre vyhodnotenie klasifikácie modelu B32,64,128. . . 36
3 – 12 Matica zámen pre vyhodnotenie klasifikácie modelu C32,64,128,256. 37
3 – 13 Matica zámen pre vyhodnotenie klasifikácie modelu D16,32,64,128. 38
3 – 14 Matica zámen pre vyhodnotenie klasifikácie modelu E32,64,128. . . 40
3 – 15 Architektúra modelu Z32,64,128. 41
3 – 16 Matica zámen pre vyhodnotenie klasifikácie modelu s attention vrstvou Z32,64,128. 42
3 – 17 Priebeh hodnôt chybových funkcií počas trénovania modelov A,B,C,D,E a Z 43
3 – 18 Ukážka nesprávne klasifikovaných obrázkov nočnej oblohy modelom B. 44
3 – 19 Rozdelenie nových snímok zo stanice Mt John do tried modelom B32,64,128. 46
3 – 20 Obrázok nočnej oblohy zo stanice Mt John klasifikovaný do triedy **clear**. 46

Zoznam tabuliek

3 – 1 Početnosti snímok nočnej oblohy podľa tried. 27
3 – 2 Rozdelenie dát na trénovaciu, validačnú a testovaciu množinu. 28
3 – 3 Vyhodnotenie modelu Random Forest. 30
3 – 4 Vyhodnotenie prvého modelu A32,64 na testovacích dátach. 33
3 – 5 Vyhodnotenie druhého modelu B32,64,128 na testovacích dátach. . . . 35
3 – 6 Vyhodnotenie tretieho modelu C32,64,128,256 na testovacích dátach. 37
3 – 7 Vyhodnotenie modelu D16,32,64,128 na testovacích dátach. 38
3 – 8 Vyhodnotenie modelu E32,64,128 na testovacích dátach. 39
3 – 9 Vyhodnotenie posledného modelu Z32,64,128 na testovacích dátach. . 41
3 – 10 Vyhodnocovacia tabuľka všetkých konvolučných modelov. 42

Zoznam symbolov a skratiek

Adam Algoritmus adaptívnej optimalizácie momenta

ASI All sky imager

CNN Convolutional Neural Network

FN False Negative

FP False Positive

LightGBM Light gradient Boosting method

MEAT Multi-Exposure Adaptive Threshold

MSE Mean Squared Error

RGB Red Green Blue

ReLU Recified Linear Unit

TN True Negative

TP True Positive

2DPCA two-dimensional principal component analysis

Úvod

Rýchly rozvoj výpočtovej techniky a dostupnosť výkonných grafických procesorov umožnili vznik nových prístupov v oblasti spracovania obrazu a strojového učenia. Hlboké učenie, ako jedna z najdynamickejšie sa rozvíjajúcich oblastí umelej inteligencie, zažíva v poslednom desaťročí výrazný pokrok najmä vďaka dostupnosti veľkých datasetov a efektívnejšiemu trénovaniu. Tento pokrok umožňuje aplikovať modely hlbokého učenia aj v oblastiach, ktoré si predtým vyžadovali manuálnu prácu odborníkov alebo špecializované riešenia.

Jednou z takýchto oblastí je aj automatická klasifikácia snímok nočnej oblohy. Celooblohové kamery produkujú rozsiahle množstvo dát, ktoré je potrebné spracovať a vyhodnotiť. Prítomnosť oblačnosti na týchto snímkach má zásadný vplyv na kvalitu ďalších pozorovaní a vedeckých meraní. Automatická detekcia oblačnosti je preto dôležitým krokom pri filtrovaní dát.

V tejto bakalárskej práci sa zameriavame na klasifikáciu snímok nočnej oblohy pomocou konvolučných neurónových sietí. Cieľom je navrhnúť modely schopné identifikovať mieru oblačnosti a klasifikovať snímku do jednej z troch kategórií: jasná, čiastočne zamračená alebo zamračená. Práca sa venuje návrhu architektúr modelov, ich trénovaniu, hodnoteniu a porovnaniu výkonnosti pomocou metrík, ako sú úspešnosť klasifikácie, presnosť, návratnosť a F1-skóre.

V prvej kapitole poskytujeme teoretický základ pre pochopenie problematiky hlbokého učenia, najmä konvolučných neurónových sietí, a vysvetľujeme základné pojmy ako aktivačné funkcie, optimalizácia či regularizácia. Druhá kapitola je venovaná celooblohovým kamerám, ktoré zachytávajú nielen oblačnosť, ale aj iné javy ako napríklad airglow či plazmové bubliny. Zároveň obsahuje prehľad súčasného stavu výskumu v oblasti spracovania snímok nočnej oblohy a podobných prác využívajúcich metódy strojového učenia.

Tretia kapitola opisuje metodológiu vývoja modelov, spracovanie dát, implementáciu architektúr a trénovanie. V závere práce analyzujeme dosiahnuté výsledky,

porovnávame výkonnosť modelov a ich potenciál pre praktické využitie v oblasti automatizovaného spracovania astronomických dát.

1 Hlboké učenie

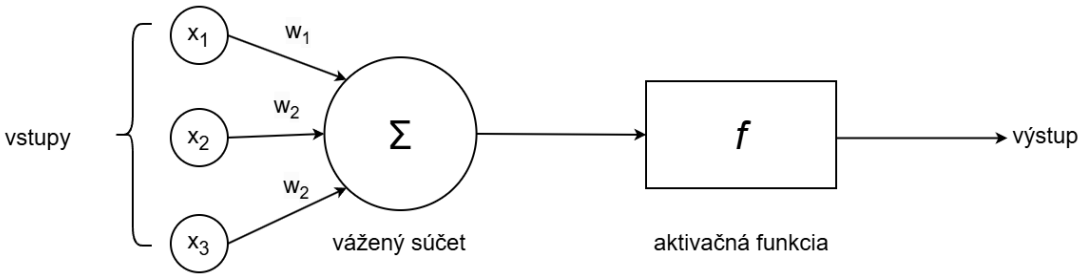
1.1 Úvod do hlbokého učenia

Hlboké učenie je jednou z najzaujímavejších a najrýchlejšie rastúcich oblastí umelej inteligencie, ktorá nám pomáha riešiť problémy, ktoré by sme ešte pred pár rokmi považovali za nemožné. V dnešnej dobe má však široké využitie – od rozpoznávania tvárí v aplikáciách, cez autonómne autá, až po generovanie textu alebo analýzu veľkých dát.

Na rozdiel od bežných algoritmov, ktoré sa riadia pevne stanovenými pravidlami, hlboké učenie umožňuje modelom učiť sa z dát, ktoré im dávame. To znamená, že modely „zistia“ sami, ako správne rozpoznať vzory alebo nájsť riešenia, bez toho, aby ich niekto musel manuálne naprogramovať.

1.2 Neurónové siete

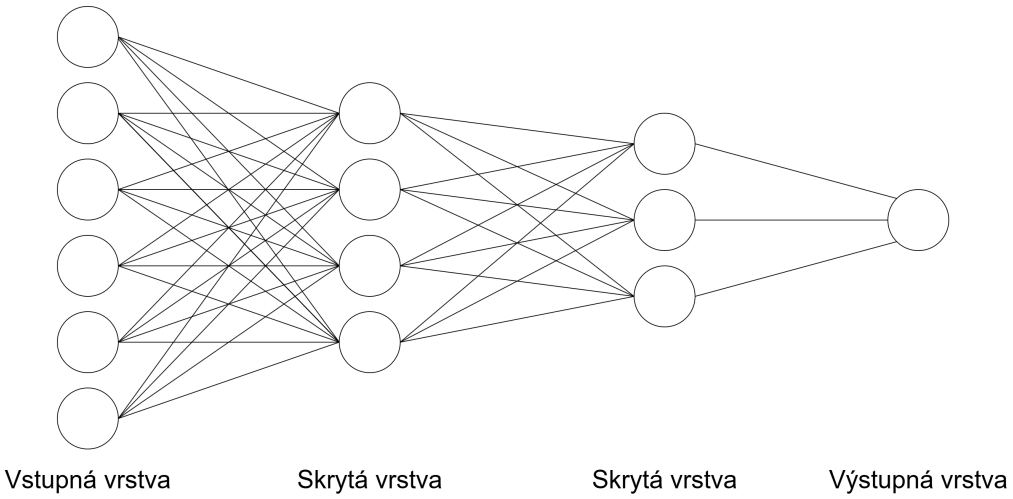
Hlavným nástrojom v hlbokom učení sú neurónové siete. Architektúra týchto sietí bola inšpirovaná štruktúrou ľudského mozgu. Základným stavebným prvkom umelej neurónovej siete je perceptrón (Obrázok 1 – 1), ktorý prijíma vstupy, ktoré sú následne vynásobené váhami a cez aktivačnú funkciu sa generuje výstup perceptrónu.



Obrázok 1 – 1 Architektúra perceptrónu.

Neurónová sieť pozostáva z viacerých typov vrstiev. Vo všeobecnosti poznáme tri základné typy (Goodfellow et al., 2016):

1. **Vstupná vrstva** – Slúži na vloženie vstupných dát do modelu.



Obrázok 1 – 2 Architektúra neurónovej siete.

2. **Skrytá vrstva** – Týchto vrstiev môže byť viacero, pričom každá z nich slúži na rozpoznávanie a extrakciu vzorcov alebo znalostí zo vstupných dát. Tieto informácie sa potom posúvajú do ďalších vrstiev v modeli.

3. **Výstupná vrstva** – Poskytuje výsledné predikcie alebo klasifikácie na základe spracovaných dát.

Na Obrázku 1 – 2 je znázornená architektúra neurónovej siete. Táto sieť využíva takzvané plne prepojené vrstvy, v ktorých je každý neurón danej vrstvy prepojený so všetkými neurónmi v nasledujúcej vrstve. V prípade, ak sa informácie v sieti šíria iba smerom dopredu, hovoríme o doprednej neurónovej sieti.

Aby bola neurónová sieť plne funkčná, je nevyhnutné správne nastaviť podmienky, za ktorých sa jednotlivé neuróny aktivujú. Samotné učenie siete prebieha pomocou algoritmu spätného šírenia chyby (angl. *backpropagation*), ktorý umožňuje upravovať váhy tak, aby sa zmenšovala chyba na výstupe. V nasledujúcich podkapitolách sa preto zameriame na ďalšie kľúčové aspekty neurónových sietí, ako sú aktivačné a chybové funkcie, optimalizácia a regularizácia učenia.

1.2.1 Aktivačné funkcie

Aktivačné funkcie sú dôležitou súčasťou neurónových sietí, pretože vykonávajú nelineárnu transformáciu a určujú, či sa konkrétny neurón aktivuje alebo nie. Bez aktivačných funkcií by bol výstup siete len lineárnou kombináciou vstupov, čo by znamenalo, že výsledný model by bol v podstate iba polynómom prvého stupňa. Lineárna rovnica je síce jednoduchá na výpočty, no jej schopnosti sú obmedzené, keď ide o riešenie komplexnejších úloh, ako je napríklad spracovanie obrazov. Preto je v každej vrstve neurónovej siete použitie aktivačných funkcií nevyhnutné na dosiahnutie požadovanej komplexnosti a získania lepších výsledkov (Sharma et al., 2017).

V nasledujúcej časti sa zameriame na aktivačné funkcie, ktoré boli použité v praktickej časti, konkrétne na ReLU, ktorá sa používala v skrytých vrstvách, a softmax, ktorá bola aplikovaná vo výstupnej vrstve.

- **ReLU**

  Funkcia ReLU (angl. *Rectified Linear Unit*) je aktivačná funkcia s nelineárnym priebehom a je definovaná takto:

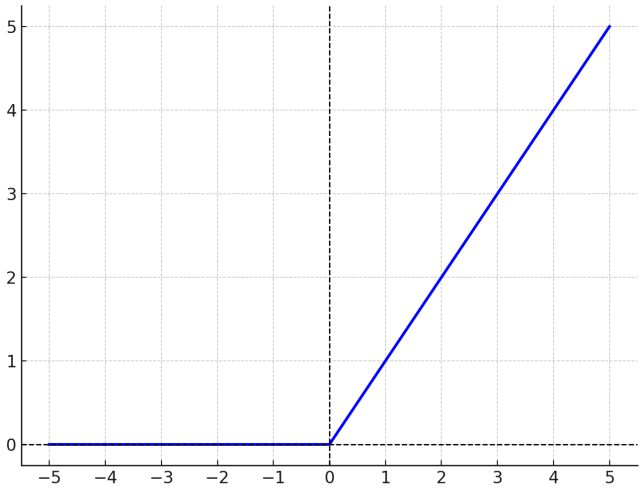
  $$f(x) = \max(0, x). \tag{1.1}$$

  Táto funkcia je v neurónových sieťach využívaná často. Jej hlavnou výhodou je, že nezapája všetky neuróny naraz. Na obrázku 1 – 3 môžeme pozorovať, že neuróny sa aktivujú iba vtedy, keď vstupná hodnota presiahne nulu. Ak je hodnota menšia alebo rovná nule, neurón zostáva deaktivovaný.

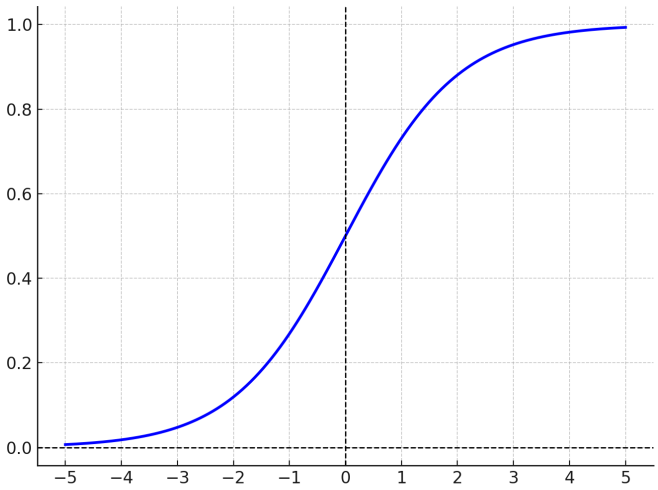
- **Sigmoid**

  Sigmoidálna funkcia (Obrázok 1 – 4) je nelineárna aktivačná funkcia, ktorá transformuje hodnoty do intervalu 0 až 1. Matematicky je funkcia Sigmoid definovaná takto:

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



Obrázok 1 – 3 Priebeh funkcie ReLU.



Obrázok 1 – 4 Priebeh funkcie Sigmoid.

Funkcia Sigmoid sa často využíva pri binárnej klasifikácii. Ak je jej výstup blízky hodnote 1, model predpokladá, že vstup patrí do jednej triedy. Ak je blízky 0, priraďuje vstup druhej triede. Hoci sme túto funkciu v našej práci nepoužili, jej pochopenie je kľúčové pre správne porozumenie aktivačnej funkcie

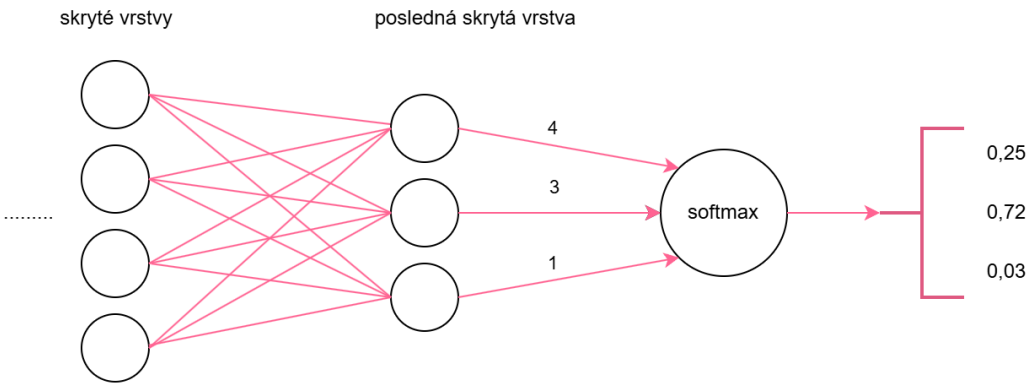
softmax, ktorú si priblížime v nasledujúcej časti.

- **Softmax**

  Softmax funkcia sa používa najmä pri úlohách viactriednej klasifikácie, kde je potrebné určiť pravdepodobnostné rozdelenie medzi viaceré triedy. Na rozdiel od sigmoidálnej funkcie, ktorá sa využíva v binárnej klasifikácii a vracia pravdepodobnosť pre jednu triedu, softmax vracia pravdepodobnosti pre každú triedu. Matematicky ju môžeme vyjadriť nasledovne:

$$f(x_i) = \frac{e^{x_i}}{\sum_j e^{x_j}}. \tag{1.3}$$

  V našom prípade sme funkciu Softmax využili v poslednej vrstve neurónovej siete na klasifikáciu, aby sme určili, do ktorej triedy patrí daný obrázok nočnej oblohy. Posledná vrstva, ako je znázornené na Obrázku 1 – 5, transformuje výstupy siete na pravdepodobnosti jednotlivých tried, pričom súčet týchto pravdepodobností je vždy rovný jednej.



Obrázok 1 – 5 Znázornenie funkcie softmax.

1.2.2 Chybové funkcie

Chybové funkcie (angl. *loss functions*) sú kľúčovým prvkom pri trénovaní neurónových sietí, pretože určujú, ako dobre alebo zle model predpovedá výsledky. Ich

úlohou je vypočítať rozdiel medzi skutočnými hodnotami a predikciami modelu, pričom cieľom trénovania je túto chybu minimalizovať a priblížiť hodnoty predikcie čo najviac k skutočným hodnotám testovacích dát. Hodnota chybovej funkcie vyjadruje, ako presne model predpovedá výsledky. Vyššia hodnota znamená väčšiu chybu pri predikcii, zatiaľ čo nižšia hodnota naznačuje, že predikcie sú presnejšie.

Existuje niekoľko rôznych typov chybových funkcií, ktoré sa používajú v závislosti od typu úlohy. Medzi najznámejšie patria stredná kvadratická chyba (angl. *Mean Squared Error*, skrátene MSE), ktorá sa používa pri regresii, binárna krížová entropia, ktorá je určená pre binárnu klasifikáciu, a kategorická krížová entropia, ktorá sa využíva pri klasifikácii do viacerých tried (Chollet and Chollet, 2021). Keďže v praktickej časti sa venujeme viactriednej klasifikácii, ďalej sa budeme venovať kategorickej krížovej entropii. Je to najbežnejšia chybová funkcia pre problémy, kde model predpovedá pravdepodobnosť každej triedy a vypočíta sa ako:

$$L = -\sum_{i=1}^{n} y_i \log(\hat{y}_i), \tag{1.4}$$

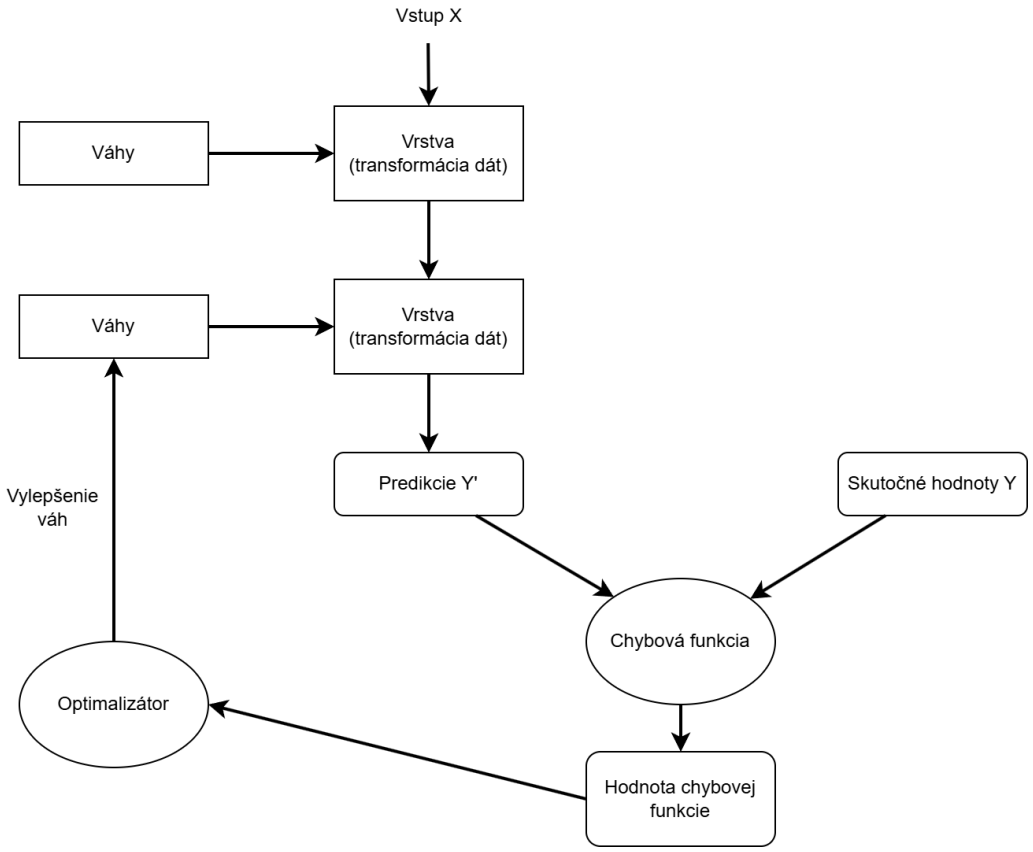
kde y_i je skutočný vektor a $\hat{y}_i$ je predikovaná pravdepodobnosť pre triedu i a n je počet tried.

1.2.3 Optimalizácia

Optimalizácia neurónových sietí je proces, ktorý sa zameriava na úpravu parametrov siete, ako sú váhy, s cieľom minimalizovať hodnotu chybovej funkcie počas trénovania. Cieľom optimalizácie je dosiahnuť čo najlepšie výstupy modelu na základe zadaných vstupov, čím sa zlepšuje jeho schopnosť generalizovať na nové, doposiaľ neznáme dáta. Na tento účel sa používajú rôzne algoritmy.

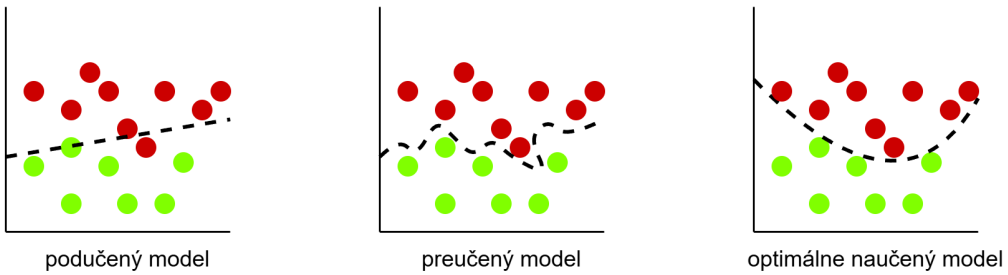
Predtým, než prejdeme k optimalizačným algoritmom, je dôležité spomenúť proces spätného šírenia chyby, ktorý je kľúčový pre trénovanie neurónových sietí. Tento algoritmus využíva hodnoty chybovej funkcie na výstupe ako spätnú väzbu, ktorá určuje, ako sa majú upraviť váhy v sieti. Na základe tejto spätnej väzby sa vykonávajú úpravy váh s cieľom minimalizovať chybu, čím sa model optimalizuje. Celý

algoritmus je znázornený na obrázku 1 – 6, kde je vidieť, ako sa modifikácie váh vykonávajú na základe chybovej hodnoty. Tento proces sa opakuje počas trénovania modelu, pričom každá iterácia prispieva k zlepšeniu schopnosti modelu vykonávať presnejšie predpovede.



Obrázok 1 – 6 Algoritmus spätného šírenie chýb.

V našom prípade sme v praktickej časti využili veľmi známy optimalizačný algoritmus Adam (Kingma and Ba, 2014), ktorý kombinuje výhody metód Adagrad (Duchi et al., 2011) a RMSprop (Tieleman, 2012). Tento algoritmus je jednoduchý na implementáciu, má nízke nároky na pamäť, prispôsobuje sa rôznym rýchlostiam učenia jednotlivých parametrov a je obzvlášť vhodný na prácu s veľkým množstvom dát.



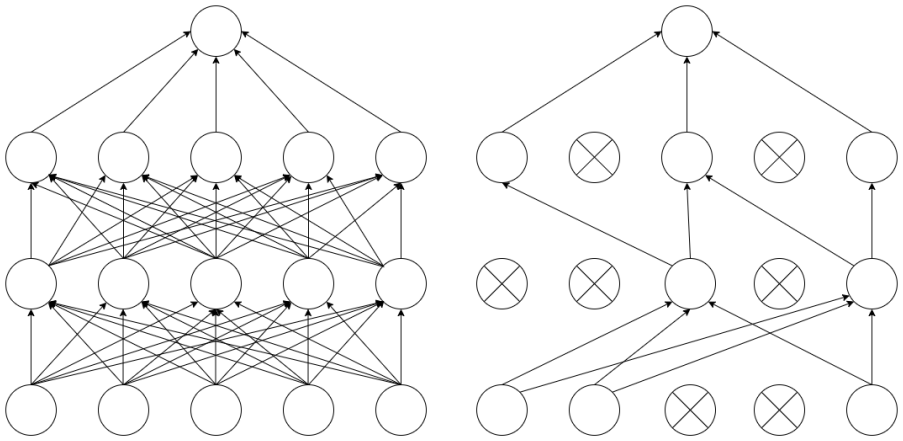
Obrázok 1 – 7 Podučenie, preučenie a optimálny model.

1.2.4 Regularizácia

Počas trénovania modelu môže nastať mnoho komplikácií. Jednou z nich je jav nazývaný preučenie (angl. *overfitting*), ktorý nastáva, keď je model príliš zameraný na špecifické črty v trénovacích dátach a nie je schopný získavať generalizované znalosti z trénovacích dát. Výsledkom je, že model dosahuje výrazne lepšie výsledky na trénovacích dátach v porovnaní s validačnými.

Riešením tohto problému môže byť úprava architektúry neurónovej siete, napríklad znížením počtu neurónov alebo úplným odstránením niektorých vrstiev. Pri týchto úpravách je však potrebné postupovať opatrne, aby sme sa vyhli opačnému extrému, teda podučeniu (angl. *underfitting*), keď je model príliš jednoduchý a nedokáže vôbec zachytiť dôležité vzory v dátach. Jednotlivé modely môžeme vidieť na Obrázku 1 – 7.

Jednou z najčastejšie využívaných techník regularizácie je metóda dropout (Obrázok 1 – 8), ktorú sme aplikovali v praktickej časti tejto práce. Táto metóda počas trénovania v každom cykle deaktivuje pevne stanovený počet neurónov, pričom ich vyberá náhodne, s cieľom minimalizovať závislosť modelu na konkrétnych neurónoch (Srivastava et al., 2014).



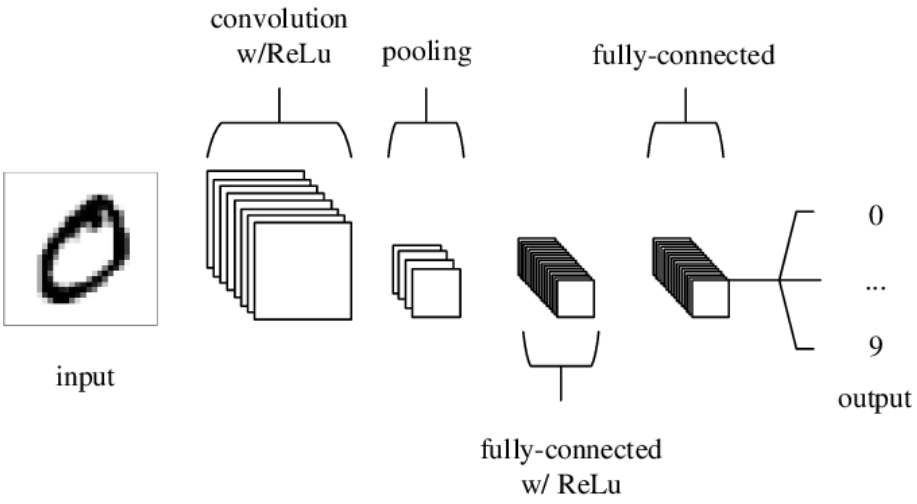
Obrázok 1 – 8 Znázornenie regularizácie pomocou metódy dropout.

1.3 Konvolučné neurónové siete

Konvolučné neurónové siete (angl. *Convolutional Neural Networks*, CNN) predstavujú špecializovaný typ neurónových sietí, ktoré sa často využívajú pri spracovaní obrazových, zvukových a textových dát. Ich hlavnou výhodou je schopnosť efektívne extrahovať významné črty z dát a zachytávať priestorové vzťahy medzi jednotlivými prvkami.

Od tradičných plne prepojených dopredných neurónových sietí sa CNN odlišujú prítomnosťou konvolučných vrstiev, ktoré umožňujú učenie sa lokálnych vzorov (Chollet and Chollet, 2021). Na obrázku 1 – 9 môžeme vidieť konvolučnú neurónovú sieť, ktorá sa skladá z nasledujúcich vrstiev:

- **Vstupná vrstva** (angl. *Input layer*), ktorá slúži na vloženie dát do siete. Napríklad pri spracovaní obrazov majú vstupné dáta tri rozmery: výška, šírka a počet kanálov (napr. RGB obrázok má tri kanály).
- **Konvolučná vrstva** (angl. *Convolutional layer*) je základná stavebná jednotka CNN, ktorá využíva súbor filtrov (angl. *kernels*) na extrakciu významných čŕt z dát.
- **Združovacia vrstva** (angl. *Pooling layer*), ktorá redukuje rozmery dát a tým



Obrázok 1 – 9 Jednoduchá konvolučná neurónová sieť (O'shea and Nash, 2015).

znižuje počet parametrov v sieti, čím pomáha predchádzať preučeniu. Bežné typy sú max pooling a average pooling.

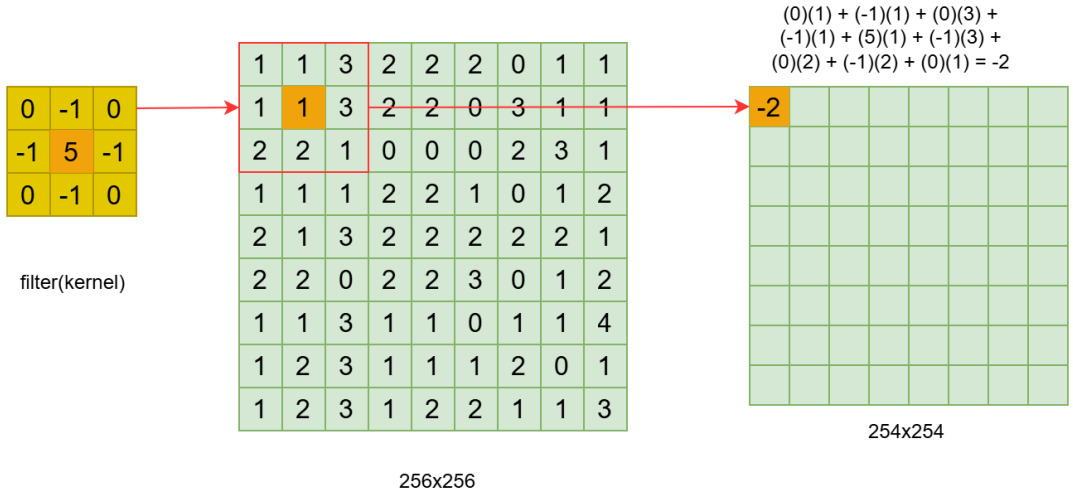
- **Plne prepojená vrstva** (angl. *Dense layer*), ktorá vykonáva konečnú klasifikáciu na základe extrahovaných čŕt.

Ako sme už spomenuli, v CNN zohráva kľúčovú úlohu konvolučná vrstva, ktorá využíva operáciu konvolúcie na extrakciu relevantných čŕt z vstupných dát, ako sú hrany, textúry alebo zložitejšie vzory.

Na Obrázku 1 – 10 môžeme vidieť názornú ukážku procesu konvolúcie. Celý proces začína filtrom, čo je malá matica, ktorá sa posúva cez vstupné dáta s určitým krokom a vykonáva na nich operáciu konvolúcie. V tomto príklade má filter veľkosť 3×3. Vstupné dáta sú reprezentované vstupnou maticou, ktorá môže zodpovedať napríklad obrazu s rozmermi 256×256 pixelov. Výsledkom tejto operácie je výstupná matica, teda mapa čŕt, ktorá má v tomto prípade rozmery 254×254. Výstupný prvok (-2) vznikne ako súčet súčinov prvkov filtra a zodpovedajúcich prvkov z podmatice vstupných dát (Goodfellow et al., 2016), podľa nasledujúceho vzorca:

$$S(i,j) = (I * K)(i,j) = \sum_m \sum_n I(m,n)K(i-m, j-n), \tag{1.5}$$

kde:

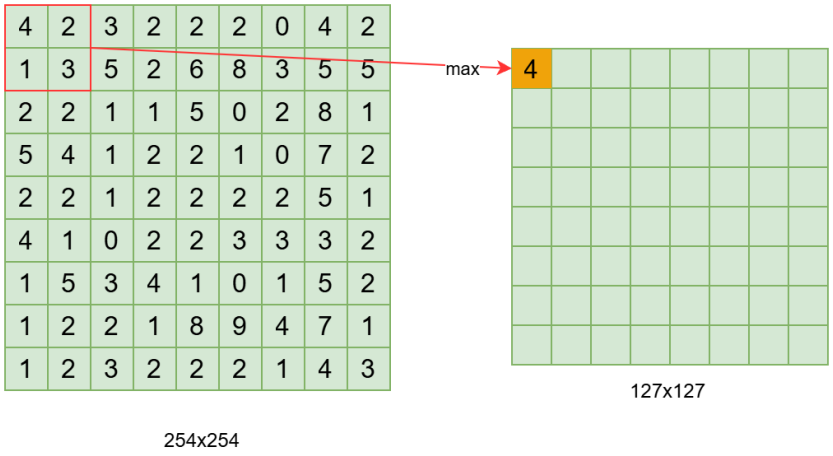


Obrázok 1 – 10 Proces konvolúcie.

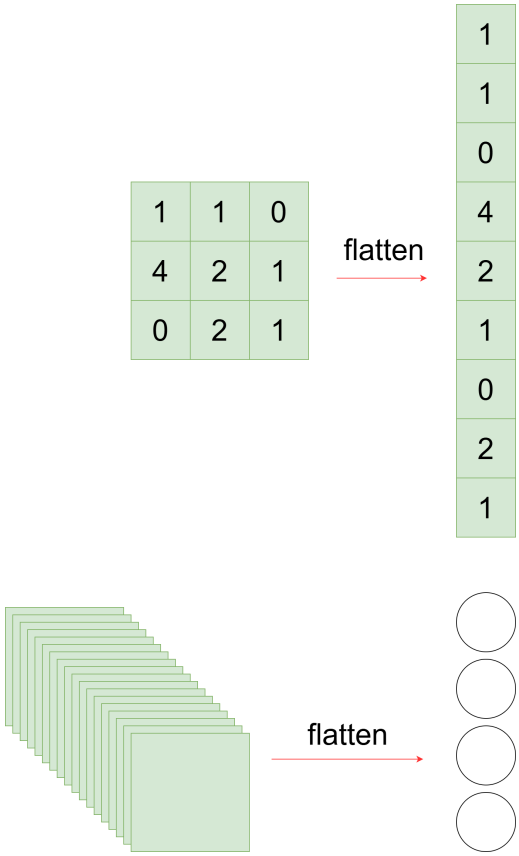
- $S(i, j)$ je výsledná hodnota v pozícii (i, j) výstupnej *feature* mapy (mapy čŕt),
- $I(m, n)$ predstavuje hodnoty vstupnej matice,
- $K(i - m, j - n)$ je hodnota filtra v príslušnej pozícii.

Aby sme ďalej znížili rozmery výstupnej mapy čŕt a zachovali najdôležitejšie informácie, využívame poolingovú vrstvu. V našom prípade sme v praktickej časti použili MaxPooling2D s oknom veľkosti 2×2, ktoré z každej takejto oblasti vyberie maximálnu hodnotu, tak ako to vidíme na Obrázku 1 – 11.

Po spracovaní vstupných dát konvolučnými a pooling vrstvami nasleduje takzvaná flatten vrstva, ktorú môžeme vidieť na Obrázku 1 – 12. Jej úlohou je previesť viacrozmerné výstupy z predchádzajúcich vrstiev do jednorozmerného vektora. Tento vektor potom slúži ako vstup pre husté (plne prepojené) vrstvy, ktoré realizujú klasifikáciu. Vďaka flatten vrstve je možné plynule prepojiť extrakciu príznakov s rozhodovacou časťou neurónovej siete.



Obrázok 1 – 11 Maxpooling vrstva.



Obrázok 1 – 12 Flatten pre jeden a viacero kernelov.

2 Airglow a jeho pozorovanie

2.1 Airglow

Počas jasných nocí, na miestach kde je svetelné znečistenie nízke, je možné pozorovať okrem hviezd aj jemné svetelné žiarenie známe ako airglow. Airglow je slabé prirodzené žiarenie atmosféry spôsobené chemickými reakciami v horných vrstvách atmosféry, predovšetkým vo výškach medzi 80 až 300 km (Solomon and Abreu, 1989). Toto žiarenie vzniká v dôsledku rekombinácie atómov a molekúl, ktoré boli počas dňa ionizované slnečným žiarením. V podstate by sme mohli povedať, že airglow vizualizuje hranicu medzi Zemou a vesmírom.

Airglow je viditeľný najmä v noci, keď je obloha tmavá, no na rozdiel od polárnej žiary (angl. *aurora*) je oveľa slabší a vyskytuje sa vo všetkých geografických šírkach, nielen v polárnych oblastiach. Tento jav je možné pozorovať nielen z povrchu Zeme, ale aj z obežnej dráhy, napríklad prostredníctvom snímok z Medzinárodnej vesmírnej stanice, ako môžeme vidieť na Obrázku 2 – 1.

2.2 Celooblohová kamera

Celooblohová kamera (angl. *all-sky imager*, ASI) je optický systém navrhnutý na snímanie celej oblohy v jednom zábere. Tento systém využíva špeciálne kamery s objektívom, ktorý má efekt rybieho oka, a ktorý umožňuje zachytiť oblohu v rozsahu približne 180° (Martinis et al., 2018).

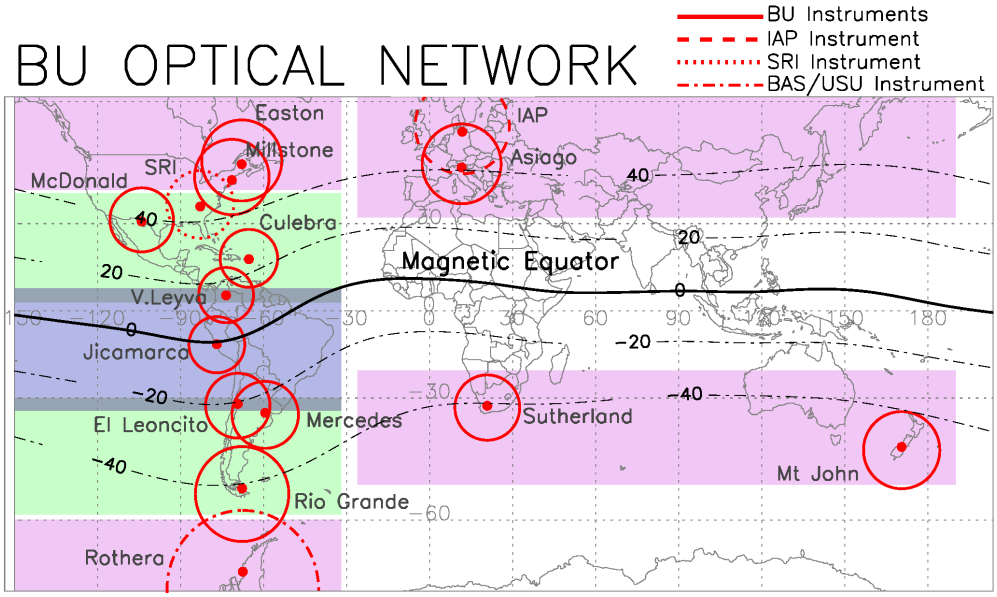
Obrázok 2 – 2 znázorňuje rozloženie celooblohových kamier, ktoré spravuje Bostonská univerzita. Tieto kamery sú rozmiestnené na rôznych zemepisných šírkach na pozorovanie ionosférických a mezosférických javov. Rozdelenie je založené na geomagnetickej šírke. Napríklad v blízkosti rovníka sa zaznamenávajú plazmové bubliny. Tieto bubliny, ktoré môžeme vidieť na Obrázku 2 – 3, sú fenoménom, ktorý vzniká v ionosfére ako dôsledok rôznych nestabilít v plazme. Tieto bubliny môžu ovplyvniť šírenie rádiových vĺn, čo môže mať nepriaznivé dopady na komunikačné



Obrázok 2 – 1 Polárna žiara a airglow pozorovaná z Medzinárodnej vesmírnej stanice ISS. Zdroj: NASA.

a navigačné systémy. Plazmové bubliny sú spojené s rôznymi dynamickými procesmi v ionosfére, vrátane vplyvu slnečnej aktivity a geomagnetických faktorov (Bhattacharyya, 2022).

Celooblohová kamera ASI využíva filtre, ktoré prepúšťajú rôzne vlnové dĺžky, čo umožňuje zachytávať rôzne emisie z hornej atmosféry. Tieto filtre sú kalibrované na vlnové dĺžky 557,7 nm, 630,0 nm, 777,4 nm, 589,3 nm, >695,0 nm a 605,0 nm alebo 644,4 nm, pričom každá z týchto vlnových dĺžok je spojená s rôznymi prvkami v atmosfére a v ionosfére, ako je kyslík a dusík (Martinis et al., 2018). V našom prípade sme na trénovanie využívali obrázky snímané pri vlnovej dĺžke 644,4 nm. Táto vlnová dĺžka bola zvolená práve preto, že v nej nie sú prítomné žiadne emisie airglowu. To z nej robí vhodný kanál na identifikáciu prítomnosti oblačnosti v spodnej atmosfére.

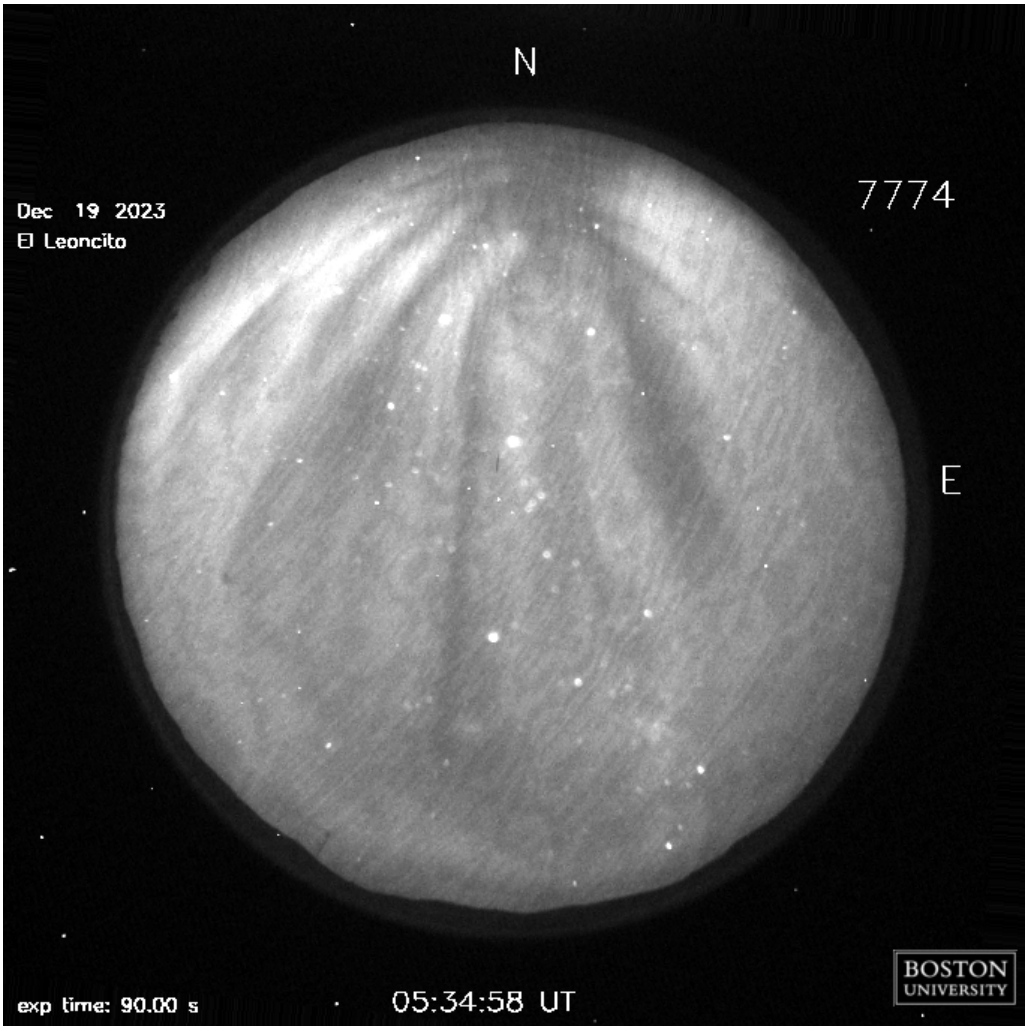


Obrázok 2 – 2 Rozmiestnenie staníc celooblohových kamier, ktoré prevádzkuje Bostnoská Univerzita. Zdroj: `https://sirius.bu.edu/aeronomy/`.

2.3 Analýza súčasného stavu

V článku Cazorla Cabrera et al. (2015), Cazorla a kolektív vyvinuli metódu s názvom Multi-Exposure Adaptive Threshold (MEAT), ktorá sa zameriava na presnejšiu detekciu oblačnosti pomocou snímok oblohy. Hlavnou myšlienkou tejto metódy je využitie série fotografií oblohy zhotovených s rôznymi nastaveniami expozície. Expozícia určuje, ako dlho je snímač fotoaparátu vystavený svetlu, a teda záleží od nej, aký jasný alebo tmavý bude výsledný obraz.

Pri bežnom fotografovaní oblohy môže nastať problém s príliš jasnými oblasťami (napríklad v okolí Slnka) alebo príliš tmavými oblasťami (v tieni oblakov). Metóda MEAT tento problém rieši tak, že kombinuje informácie z viacerých snímok. Snímky s krátkou expozíciou zachytia detaily v jasných oblastiach, zatiaľ čo snímky s dlhou expozíciou odhalia detaily v tmavých oblastiach. Následne sa na každú z týchto snímok aplikuje adaptívne prahovanie, čo je technika, ktorá automaticky určuje, ktoré pixely patria k oblakom a ktoré k jasnej oblohe, na základe lokálnych vlastností obrazu.



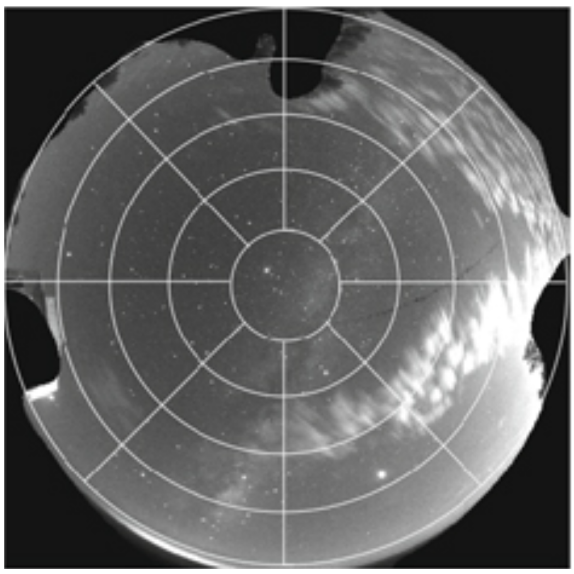
Obrázok 2 – 3 Plazmová bublina na snímke zo stanie EL Leoncito 19. decembra 2023.

Autori zistili, že z hľadiska úspešnosti MEAT dokáže pomerne spoľahlivo identifikovať prítomnosť oblačnosti. V experimentoch dosiahla metóda úspešnosť približne 90% pri detekcii prípadov, keď bolo Slnko zakryté oblakmi. Pri rozlišovaní jasnej oblohy boli výsledky o niečo horšie, s dosiahnutou úspešnosťou okolo 78%. Je dôležité poznamenať, že táto metóda sa ukázala ako obzvlášť efektívna pri spracovaní snímok s vysokým dynamickým rozsahom, teda takých, kde sú veľké rozdiely medzi najjasnejšími a najtmavšími oblasťami.

S rovnakými dátami pracovali vo svojom článku aj Yacoub et al. (2025), kde sa autori zaoberali automatickou detekciou plazmových bublín pomocou strojového učenia. Na spracovanie údajov využili dvojrozmernú analýzu hlavných komponentov (2DPCA) v kombinácii s rekurzívnou elimináciou príznakov (RFE) a modelom Random Forest. Dataset obsahoval 2458 snímok ASI rozdelených na dve triedy: „Empty“ (bez plazmovej bubliny) a „Event“ (s plazmovou bublinou), pričom rozdelenie snímok bolo v pomere 80% na trénovanie a 20% na testovanie. Pri použití samotnej 2DPCA dosiahli presnosť 97,96%. Po aplikácii Radon transformácie (integrálna transformácia obrazu) sa síce výpočtová náročnosť modelu znížila o približne 65%, avšak presnosť mierne klesla na 96,1%. Finálny XAI model kombinujúci tieto prístupy následne dosiahol až 98,17% presnosť, pričom prekonal aj porovnateľné hlboké neurónové siete ako ResNet18, ktorá dosiahla maximálne 91,45%. Výhodou tohto prístupu bola taktiež výrazne nižšia doba trénovania.

V článku Mommert (2020) autor predstavuje prístup k automatickej detekcii oblačnosti na snímkach z celooblohových kamier pomocou metód strojového učenia. Cieľom jeho práce bolo vyvinúť spoľahlivý systém na včasné varovanie pred prítomnosťou oblakov, ktoré môžu negatívne ovplyvniť astronomické pozorovania. Dôležité je poznamenať, že autor riešil len binárnu klasifikáciu, určoval teda, či snímka obsahuje oblačnosť, alebo nie.

V rámci štúdie boli porovnané dva prístupy k identifikácii a lokalizácii oblakov. Prvým bola metóda hlbokého učenia založená na adaptácii reziduálnej neurónovej siete (ResNet-18), ktorá pracovala priamo s orezanými a maskovanými snímkami



Obrázok 2 – 4 Vizualizácia rozdelenia snímky z celooblohovej kamery v práci Mommert (2020).

z celooblohovej kamery pri Discovery Channel Telescope. Na trénovanie tejto siete bolo použitých približne 2 000 manuálne označených snímok. Pre spracovanie obrazových dát autor najprv orezal oblasť obsahujúcu kruhový obraz celej oblohy a následne zamaskoval časti, ktoré nezobrazovali oblohu (napríklad stromy na horizonte). Dôležitým krokom bolo rozdelenie každého obrazu na 33 menších častí. Centrálnou časťou bol kruh v strede obrazu, ktorý zodpovedal oblasti priamo nad teleskopom. Okolo tohto stredového kruhu boli vytvorené tri ďalšie kruhové pásy. Každý z týchto kruhových pásov bol následne rozdelený na osem rovnakých častí (Obrázok 2 – 4). Toto rozdelenie zohľadňovalo dôležitosť detekcie oblakov v rôznych častiach oblohy. Samotný model ResNet bol upravený tak, aby prijímal jednokanálové (čiernobiele) obrazy s kernelom o veľkosti 16×16 pixelov a produkoval výstupný vektor dĺžky 33, pričom každá zložka tohto vektora reprezentovala predikciu prítomnosti oblaku v jednej z 33 definovaných oblastí. Tento prístup dosiahol presnosť 85%, avšak vyžadoval vyššiu výpočtovú náročnosť a dlhší čas trénovania (približne 41 minút pre 10 epoch).

Druhým riešením bol model LightGBM, ktorý využíval extrahované lokálne príznaky z približne 1 000 snímok. Pre každý z 33 definovaných subregiónov v každom

obraze bolo vypočítaných 16 rôznych príznakov (napr. rozdiely hodnôt so snímkami z pred 3 a 15 minút). Tento prístup dosiahol vyššiu presnosť – až 95%, pričom bol zároveň výpočtovo efektívnejší, s časom trénovania len 12 sekúnd. Autor preto odporúča LightGBM ako praktické riešenie pre nasadenie v reálnom čase, najmä s ohľadom na jeho presnosť a efektívnosť aj pri menšom tréningovom datasete.

Autori sa v článku Mackovjak et al. (2025) venovali hybridnému prístupu pre určovanie bezoblačných podmienok z ASI. Ich metóda kombinuje rozšírenú tradičnú metódu založenú na výpočte indexu jasnosti (Clearness index) s novou metódou využívajúcou CNN. Kým tradičná metóda hodnotí jas hviezd pre určenie jasnosti oblohy, CNN je navrhnutá na klasifikáciu samotných obrázkov do kategórií oblačnosti.

Kľúčovým prínosom ich práce bolo riešenie výzvy spojené s potrebou rozsiahlych označených dát pre tréning CNN. Autori demonštrovali, že tradičná metóda založená na indexe jasnosti môže efektívne slúžiť na automatické označovanie dát. Konkrétne, dáta z roku 2022 boli klasifikované touto tradičnou metódou a následne slúžili ako tréningový súbor pre ich novú CNN. Táto stratégia umožnila efektívne pripraviť tréningové dáta pre model hlbokého učenia.

Následne trénovaná CNN bola aplikovaná na rozsiahly 18-ročný dátový súbor z observatória El Leoncito, pokrývajúci obdobie od roku 2006 do roku 2023. V rámci tejto analýzy bolo identifikovaných celkovo 40 250 bezoblačných nočných hodín. Tento objem dát predstavuje významnú databázu časových intervalov bez oblačnosti, ktoré sú vhodné pre ďalšie astronomické a airglow vedecké štúdie.

3 Využitie strojového učenia pre filtrovanie dát z kozmickej fyziky

Predmetom tejto kapitoly je aplikácia metód strojového učenia na spracovanie obrazových dát z kozmickej fyziky, konkrétne na filtrovanie snímok nočnej oblohy. Cieľom tejto práce je vytvoriť model, ktorý dokáže efektívne triediť takéto snímky.

V mnohých oblastiach kozmického výskumu, ako je napríklad štúdium ionosféry a atmosférických javov, je práca s dátami vysokej kvality kľúčová. Snímky oblohy často obsahujú rušivé elementy, ako napríklad oblačnosť, ktorá môže skresľovať merania a analýzy. Preto je nevyhnutné pred ďalším spracovaním dát identifikovať a vyfiltrovať snímky, na ktorých je jasná obloha. Tento krok zabezpečuje, že následné analýzy sú založené na relevantných a presných dátach.

V tejto práci bol pri vývoji modelu použitý štruktúrovaný prístup, ktorý pozostáva z nasledujúcich kľúčových krokov: pochopenie problému, analýza dát, príprava dát, modelovanie a vyhodnotenie. Na implementáciu a trénovanie modelov strojového učenia bol použitý programovací jazyk Python a jeho rozsiahly ekosystém knižníc. Na manipuláciu s dátami a ich transformáciu boli použité knižnice ako NumPy a Pandas. Vizualizáciu dát a výsledkov umožnila knižnica Matplotlib. Pre implementáciu a trénovanie modelov neurónových sietí boli použité knižnice Keras a TensorFlow, ktoré patria k popredným platformám pre hlboké učenie.

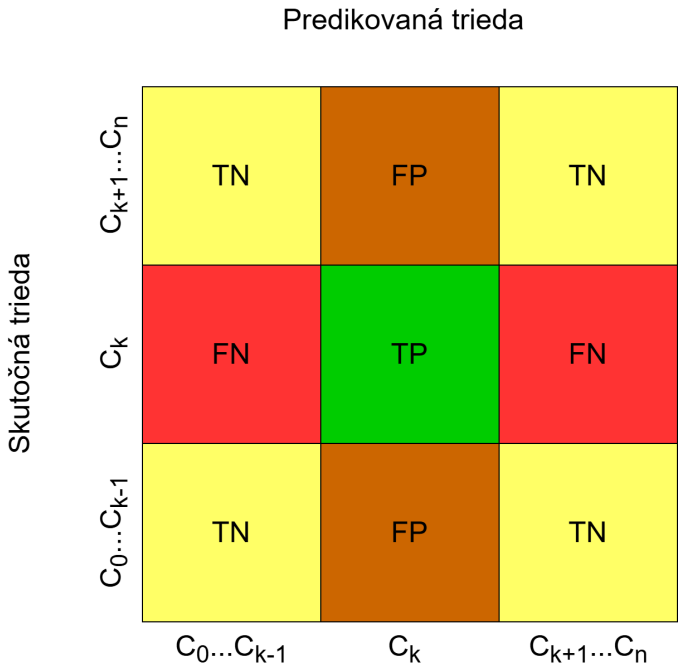
3.1 Pochopenie cieľa

Hlavným cieľom tejto bakalárskej práce je vytvorenie a vyhodnotenie modelov neurónových sietí na klasifikáciu snímok nočnej oblohy podľa stupňa oblačnosti. Ide o úlohu viactriednej klasifikácie, pri ktorej sa každému obrázku priradí jedna z niekoľkých definovaných kategórií podľa miery oblačnosti.

V rámci práce budeme experimentovať s rôznymi modelmi strojového učenia od jednoduchších algoritmov ako rozhodovací strom, logistická regresia či náhodný

les, až po hlboké konvolučné neurónové siete. Cieľom je porovnať ich výkonnosť a identifikovať najvhodnejší prístup pre danú úlohu.

Ako chybovú funkciu (*loss*) používame kategorickú krížovú entropiu, ktorá je štandardom pre úlohy viac-triednej klasifikácie s výstupom vo forme pravdepodobností. Túto funkciu sme opisovali v podkapitole 1.2.2. Vyhodnotenie jednotlivých modelov bude pozostávať z matice zámen (angl. *confusion matrix*) (Krstinić et al., 2020) a vybraných metrík, ktoré sú popísané nižšie. Najprv ale na Obrázku 3 – 1 ukážeme prípady, ktoré môžu nastať pri porovnaní skutočných a predikovaných tried pri viactriednej klasifikácii:



Obrázok 3 – 1 Matica zámen pre viactriednu klasifikáciu.

- **True Positive (TP)** - trieda C_k je správne klasifikovaná ako C_k,
- **False Negative (FN)** - iná trieda ako C_k je chybne klasifikovaná ako trieda C_k,
- **False Positive (FP)** - trieda C_k je chybne klasifikovaná ako iná trieda ako C_k,

- **True Negative (TN)** - iná trieda ako C_k je klasifikovaná ako iná trieda ako C_k.

Na základe týchto prípadov môžeme následne vypočítať pre všetky modely tieto vybrané štandardné metriky: úspešnosť klasifikácie (angl. *accuracy*), presnosť (angl. *precision*), návratnosť (angl. *recall*) a f1 skóre (angl. *f1 score*) ktoré sa vypočítajú podľa nasledujúcich vzorcov:

$$Accuracy = \frac{\sum_{k=1}^{n} TP_{C_k}}{N}, \tag{3.1}$$

kde N je počet všetkých príkladov a n je počet tried,

$$Precision = \frac{TP}{TP + FP}, \tag{3.2}$$

$$Recall = \frac{TP}{TP + FN}, \tag{3.3}$$

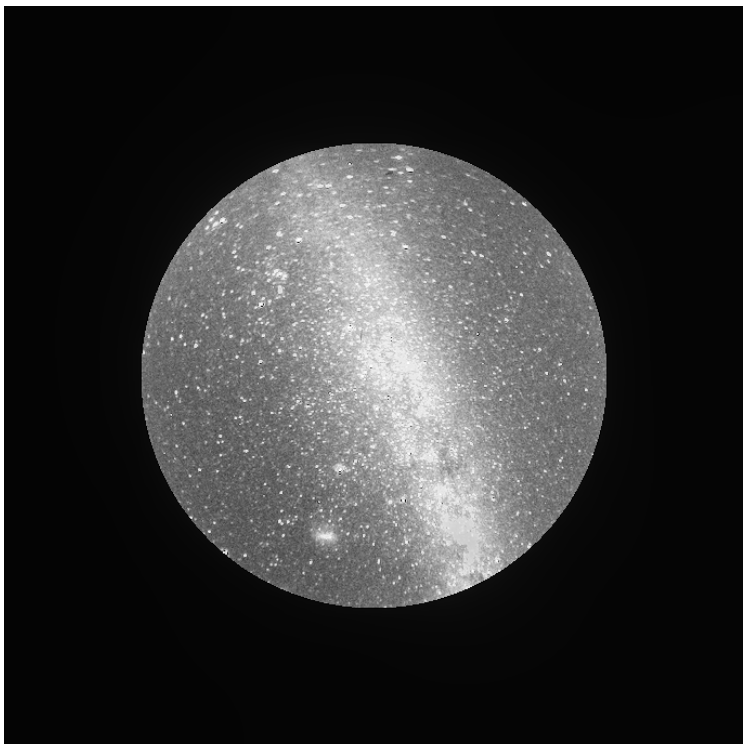
$$F1 = \frac{2 \times Precision \times Recall}{Precission + Recall}. \tag{3.4}$$

Okrem toho budeme sledovať aj priebeh *accuracy* a *loss* na validačnej množine, na základe ktorých boli modely počas tréningu optimalizované.

Primárnym cieľom teda nie je dosiahnuť konkrétnu cieľovú úspešnosť klasifikácie, ale skôr preskúmať výkonnosť rôznych modelov pri riešení tejto úlohy klasifikácie obrázkov nočnej oblohy a vybrať čo najefektívnejšie riešenie.

3.2 Pochopenie dát

V praktickej časti tejto práce pracujeme s obrazovými dátami nočnej oblohy, ktoré boli získané pomocou celooblohovej kamery prevádzkovanej Bostonskou univerzitou v areáli Complejo Astronómico El Leoncito v Argentíne. Tieto dáta sú verejne



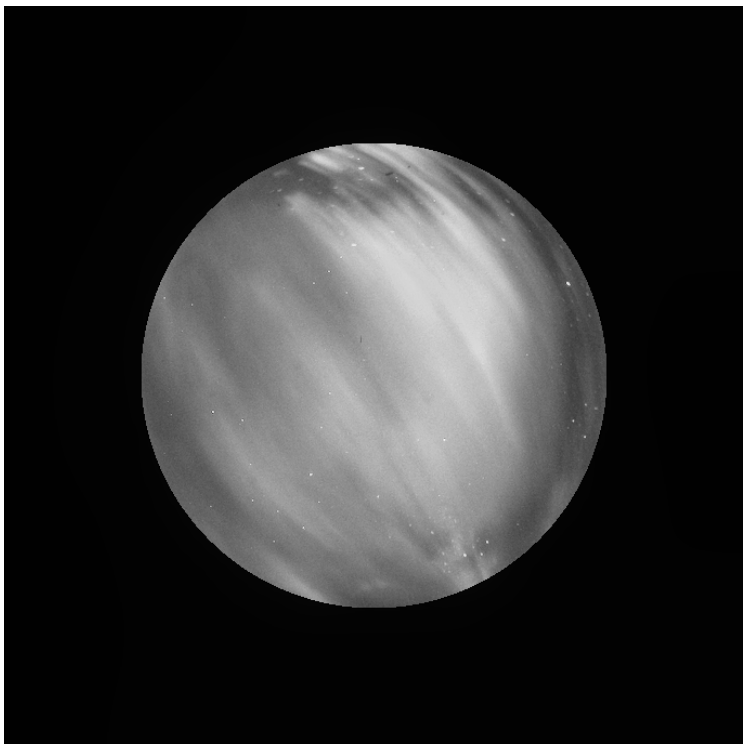
Obrázok 3 – 2 Jasná nočná obloha (trieda **clear**).

dostupné na webovej stránke projektu SIRIUS All-Sky Imager Data Archives [1] a pochádzajú z pozorovaní realizovaných v roku 2009, konkrétne v spektre s centrálnou vlnovou dĺžkou 644,4 nm.

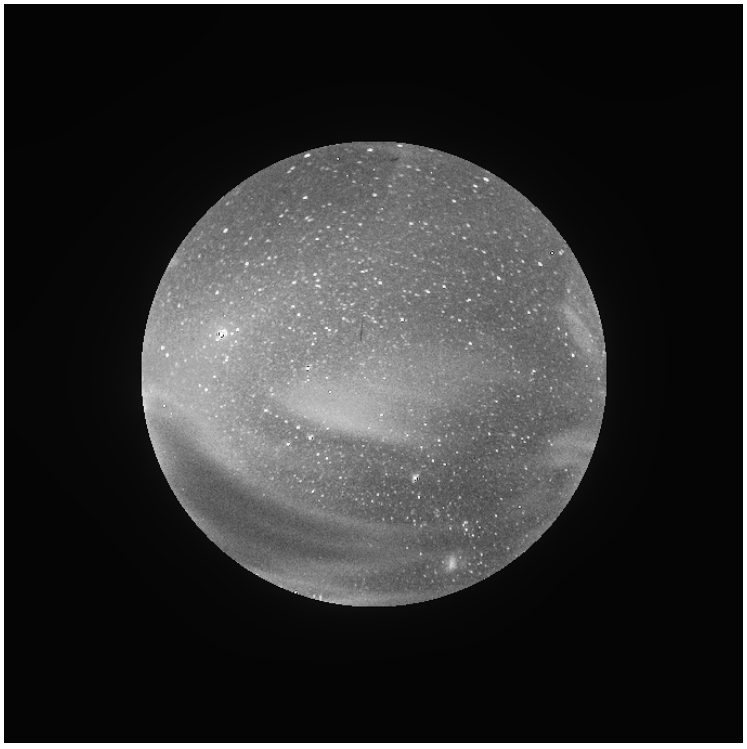
Každý obrázok nočnej oblohy je uložený vo formáte PNG s rozmermi 682×682 pixelov. Náš dataset, s ktorým budeme ďalej pokračovať, obsahuje celkovo 4 152 snímok, ktoré sú rozdelené do troch kategórií podľa miery oblačnosti: jasná obloha (angl. *clear*), čiastočne zamračené (angl. *partly cloudy*) a zamračené (angl. *cloudy*). Každá kategória je reprezentovaná osobitným priečinkom, konkrétne: `0_cloudy`, `1_partly` a `2_clear`. Príklad, ako približne vyzerá obrázok z každej kategórie, môžeme pozorovať na Obrázkoch 3 – 2, 3 – 3 a 3 – 4.

Početnosti snímok jednotlivých tried sú uvedené v nasledujúcej Tabuľke 3 – 1.

[1] Dostupné na `https://sirius.bu.edu/data/`



Obrázok 3 – 3 Zamračená nočná obloha (trieda **cloudy**).



Obrázok 3 – 4 Čiastočne zamračená nočná obloha (trieda **partly**).

Trieda	Označenie	Počet obrázkov
Zamračené (**cloudy**)	0	1 118
Čiastočne zamračené (**partly**)	1	1 316
Jasná obloha (**clear**)	2	1 718
Spolu	-	4 152

Tabuľka 3 – 1 Početnosti snímok nočnej oblohy podľa tried.

3.3 Príprava dát

Na začiatku boli všetky obrázky načítané z priečinkov podľa ich triedy a následne náhodne premiešané pomocou príkazu `np.random.shuffle`, aby sa predišlo systematickému poradiu na vstupe pri trénovaní modelu. Každý obrázok mal pôvodné rozmery 682×682 pixelov, pričom na jeho okrajoch sa nachádzali výrazné čierne pásy, ktoré môžeme vidieť na Obrázku 3 – 4. Tie boli odstránené orezaním obrázka o 100 pixelov z každej strany pomocou príkazu `reshape`.

Po orezaní boli obrázky zmenšené na rozmer 256×256 pixelov pomocou príkazu `resize`, čím sa výrazne znížila výpočtová náročnosť modelu bez výraznej straty vizuálnej informácie. Tieto predspracované obrázky boli následne prevedené na číselné polia a normalizované na interval (0, 1), teda hodnoty jednotlivých pixelov, ktoré sa pôvodne nachádzali v rozsahu od 0 do 255, boli vydelené 255, aby sa dostali do požadovaného rozsahu od 0 po 1. Takáto normalizácia je štandardným krokom pri spracovaní obrazových dát, pretože uľahčuje trénovanie modelu a zrýchľuje konvergovanie učenia vďaka jednotnému rozsahu dát.

Napokon bolo pole s obrázkami upravené do formátu, ktorý vyžaduje konvolučná neurónová sieť. Pomocou príkazu `reshape` sa pôvodný tvar dát (4 152, 256, 256) rozšíril o poslednú dimenziu, ktorá označuje počet farebných kanálov. Keďže ide o čiernobiele snímky, počet kanálov je 1, a výsledný tvar vstupných dát je teda (4 152, 256, 256, 1).

Po predspracovaní dát boli všetky obrázky náhodne rozdelené do trénovacej a tes-

tovacej množiny v pomere 80:20 pomocou funkcie `train_test_split`. Toto rozdelenie zabezpečuje, že model je trénovaný iba na časti dostupných dát, zatiaľ čo druhá časť ostáva „nevidená“ a slúži na neskoršie overenie jeho výkonnosti. Následne boli trénovacie dáta opäť rozdelené pomocou `train_test_split`, tentokrát v pomere 90:10, na samotné trénovacie a validačné dáta, pričom práve validačné dáta slúžili počas učenia modelu na priebežné vyhodnocovanie jeho výkonu. Finálne početnosti jednotlivých tried v dátových množinách sú uvedené v Tabuľke 3 – 2.

Trieda	**Trénovacie dáta**	**Validačné dáta**	**Testovacie dáta**
0 - **cloudy**	790	81	247
1 - **partly**	929	114	273
2 - **clear**	1 269	138	311
Spolu	2 988	333	831

Tabuľka 3 – 2 Rozdelenie dát na trénovaciu, validačnú a testovaciu množinu.

Po rozdelení dát do jednotlivých množín nasledovala dátová augmentácia, ktorá bola aplikovaná iba na trénovaciu množinu. Cieľom augmentácie je zvýšiť variabilitu trénovacích dát bez potreby získavania nových obrázkov. V konečnom dôsledku je možné týmto krokom zväčšiť objem trénovacej množiny. Augmentáciou sa znižuje riziko preučenia modelu, zároveň sa zvyšuje jeho robustnosť a zlepšuje jeho schopnosť extrahovania dostatočne generalizovaných príznakov. Augmentácia teda prispieva k vytvoreniu modelu, ktorý je menej citlivý na šum a variácie v reálnych vstupoch. Pri aplikovaní augmentácie na naše dáta boli využité techniky ako horizontálne a vertikálne preklápanie a náhodná rotácia obrázkov v rozsahu $\pm 45°$ zabezpečené funkciou `ImageDataGenerator` z knižnice `keras` a nastavením jej parametrov `horizontal_flip`, `vertical_flip` a `rotation_range=45`. Validačné a testovacie množiny ostali nezmenené, aby sa zabezpečilo objektívne vyhodnotenie výkonnosti modelu a finálne testovanie na predtým modelom nikdy nevidených dátach.

3.4 Modelovanie

Modely boli trénované na trénovacej množine a vyhodnocované pomocou testovacej množiny, ktorá bola oddelená pri predspracovaní dát. Hlavnou metrikou, ktorou porovnávame jednotlivé modely, je *accuracy* – teda podiel správne klasifikovaných prípadov. Hodnota tejto metriky sa pohybuje v rozsahu od 0 do 1, pričom hodnota 1 zodpovedá 100% správnosti predikcií a predstavuje ideálny výsledok.

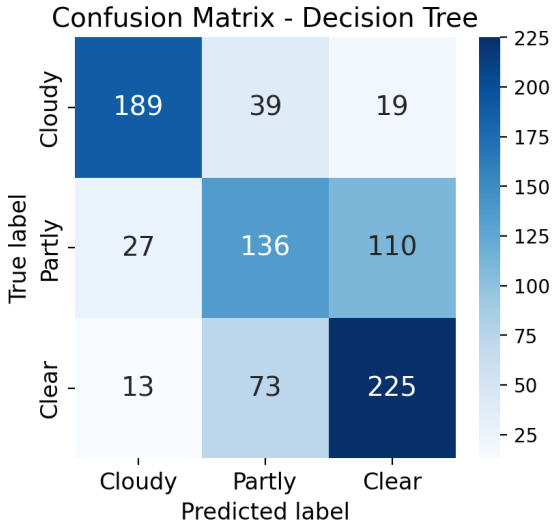
Všetky modely boli trénované v jazyku Python v prostredí Jupyter Notebooku na vlastnom počítači s grafickou kartou RTX 4070 Super.

3.4.1 Referenčné modely

Po úspešnej príprave a rozdelení dát môžeme pristúpiť k samotnému modelovaniu. Ako prvý krok boli vytvorené a otestované jednoduché klasifikačné modely, ktoré slúžia ako referenčná báza pre neskoršie porovnanie s komplexnejšími architektúrami. Využité boli základné algoritmy dostupné v knižnici `scikit-learn`, konkrétne:

- Rozhodovací strom (angl. *Decision Tree Classifier*) je model, ktorý vytvára štruktúru rozhodovacích pravidiel na základe vstupných údajov.
- Logistická regresia (angl. *Logistic Regression*) je lineárny model, ktorý predpovedá pravdepodobnosť výskytu určitej triedy pomocou logistickej funkcie.
- Náhodný les (angl. *Random Forest*) je model založený na množstve rozhodovacích stromov, kde každý strom pracuje s náhodne vybranou podmnožinou dát.

Po natrénovaní referenčných modelov sme vyhodnotili ich výkonnosť na testovacej množine. Model rozhodovacieho stromu dosiahol úspešnosť klasifikácie 0,66, logistická regresia 0,76 a najlepšie si viedol random forest classifier s úspešnosťou 0,82. Tieto výsledky ukazujú, že aj jednoduché modely môžu poskytnúť pomerne spoľahlivú klasifikáciu, pričom kombinovanie viacerých stromov v rámci random fo-



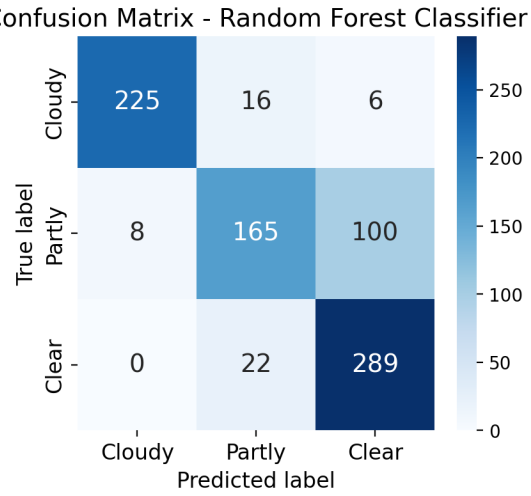
Obrázok 3 – 5 Matica zámen pre vyhodnotenie klasifikácie rozhodovacieho stromu.

restu výrazne zlepšuje výkon. V Tabuľke 3 – 3 môžeme vidieť hodnoty metrík pri vyhodnotení modelu Random Forest.

Trieda	**Presnosť**	**Návratnosť**	**f1 skóre**
0 - **cloudy**	0,97	0,91	0,94
1 - **partly**	0,81	0,60	0,69
2 - **clear**	0,73	0,93	0,82
makro priemer	0,84	0,81	0,82
vážený priemer	0,83	0,82	0,81

Tabuľka 3 – 3 Vyhodnotenie modelu Random Forest.

Pre lepšiu ilustráciu boli vytvorené matice zámen klasifikácie pre rozhodovací strom (Obrázok 3 – 5) a Random Forest (Obrázok 3 – 6). Rozhodovací strom často zamieňal triedu **partly** za **clear**, hoci triedu **cloudy** klasifikoval pomerne dobre. Random forest dosiahol lepšie výsledky, najmä pri triede **clear**, no rovnako ako rozhodovací strom mal problémy so správnym rozlíšením triedy **partly**.



Obrázok 3 – 6 Matica zámen pre vyhodnotenie klasifikácie modelu Random Forest.

3.4.2 Konvolučné neurónové siete

Po referenčných modeloch sme sa zamerali na trénovanie vlastných CNN, ktoré sú vhodnejšie na spracovanie obrazových dát. Pri ich trénovaní sme nastavovali niekoľko dôležitých parametrov. Parameter `BATCH_SIZE` určuje počet obrázkov, ktoré model spracuje naraz počas jednej iterácie. Naše experimenty prebiehali s dávkami veľkosti 32 alebo 64, čo sa ukázalo ako rozumný kompromis medzi rýchlosťou a stabilitou učenia. Počet epoch sme na začiatku nastavili na 200, aby mal model dostatočný priestor na učenie. Aby však nedošlo k preučeniu, využili sme techniku `EarlyStopping`. Tá sleduje vývoj hodnoty chybovej funkcie na validačnej množine a ak sa počas stanoveného počtu epoch nezlepší, trénovanie sa automaticky ukončí.

Model A32,64

Keď sme mali všetko pripravené, mohli sme pristúpiť k trénovaniu prvého modelu. Návrh architektúry tohto modelu je jednoduchý, pozostáva z dvoch konvolučných vrstiev s aktivačnou funkciou `relu`, kde prvá vrstva obsahuje 32 filtrov a druhá vrstva 64 filtrov, ako môžeme vidieť na Obrázku 3 – 7. Jeden filter má veľkosť 3×3 a za každou konvolučnou vrstvou nasleduje `MaxPooling2D` vrstva, ktorá slúži na

Layer (type)	Output Shape	Param #
conv2d_2 (Conv2D)	(None, 254, 254, 32)	320
max_pooling2d_2 (MaxPooling 2D)	(None, 127, 127, 32)	0
dropout_3 (Dropout)	(None, 127, 127, 32)	0
conv2d_3 (Conv2D)	(None, 125, 125, 64)	18496
max_pooling2d_3 (MaxPooling 2D)	(None, 62, 62, 64)	0
dropout_4 (Dropout)	(None, 62, 62, 64)	0
flatten_1 (Flatten)	(None, 246016)	0
dense_2 (Dense)	(None, 128)	31490176
dropout_5 (Dropout)	(None, 128)	0
dense_3 (Dense)	(None, 3)	387

Obrázok 3 – 7 Architektúra prvého jednoduchého modelu A32,64.

zmenšenie rozmerov výstupných dát. Pre zvýšenie robustnosti modelu a zníženie preučenia je po prvej `MaxPooling2D` vrstve použitá vrstva `Dropout` s nastaveným vypínaním neurónov na hodnotu 0,2, a po druhej `MaxPooling2D` vrstve ďalšia `Dropout` vrstva s parametrom 0,3, po ktorej nasleduje `Flatten` vrstva, ktorá prevádza výstup z predchádzajúcich vrstiev do jednorozmerného poľa. Nasleduje štandardná (`Dense`) vrstva s 128 neurónmi. Po nej je opäť použitá `Dropout` vrstva s hodnotou 0,4 na ďalšie zníženie rizika preučenia. Nakoniec je pridaná posledná `Dense` vrstva s tromi výstupmi, kde každý výstup reprezentuje jednu triedu (**clear**, **partly** alebo **cloudy**). Na výstupe je použitá aktivačná funkcia `softmax`, ktorá zaručuje, že výstupy modelu budú pravdepodobnosti pre každú triedu, pričom vyberáme triedu s najvyššou hodnotou ako výslednú predikciu.

Trénovanie prvého jednoduchého modelu nebolo časovo náročné. Najlepší model bol dosiahnutý v epoche 38. Chybová funkcia na validačnej množine, teda *val_loss*, dosiahla hodnotu 0,40, na základe čoho bol vybraný najlepší model.

Následne sme model otestovali na testovacích dátach. V tabuľke 3 – 4 sú uve-

Trieda	Presnosť	Návratnosť	f1 skóre
0 - **cloudy**	0,98	0,96	0,97
1 - **partly**	0,86	0,48	0,62
2 - **clear**	0,68	0,96	0,80
makro priemer	0,84	0,80	0,79
vážený priemer	0,80	0,80	0,79

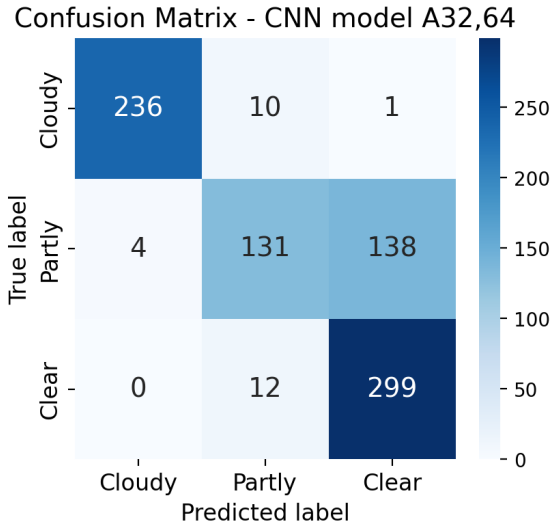
Tabuľka 3 – 4 Vyhodnotenie prvého modelu A32,64 na testovacích dátach.

dené jednotlivé metriky použité pri vyhodnotení jeho výkonnosti. Model mal pri testovacích dátach úspešnosť klasifikácie 0,80. Už tento jednoduchý model dosahuje porovnateľnú úspešnosť ako natrénovaný model Random Forest.

Na Obrázku 3 – 8 je znázornená matica zámen pre tento model. Je vidieť, že model spoľahlivo rozpoznáva jasné a úplne zamračené obrázky. Problém nastáva pri čiastočne zamračených prípadoch, ktoré často klasifikuje ako jasné. Tento nedostatok súvisí s obmedzenou kapacitou modelu, ktorý zatiaľ nedokáže správne odlíšiť jemné rozdiely medzi miernou oblačnosťou a čistou nočnou oblohou. V ďalšom kroku preto rozšírime architektúru o ďalšiu konvolučnú vrstvu a overíme, či to zlepší jeho výkonnosť.

Model B32,64,128

Ďalší model, ktorý sme vytvorili, je veľmi podobný tomu prvému, avšak s jedným podstatným vylepšením. Do architektúry sme doplnili tretiu konvolučnú vrstvu s 128 filtrami, čo by malo modelu pomôcť lepšie extrahovať zložitejšie črty z obrázkov. Zároveň sme upravili aj veľkosť plne prepojenej vrstvy, ktorá bude mať namiesto pôvodných 128 až 256 neurónov. Tieto zmeny by mali zvýšiť kapacitu modelu a jeho schopnosť rozlíšiť náročnejšie prípady, najmä pri triede **partly**. Architektúru tohto modelu je možné vidieť na Obrázku 3 – 9. Ďalším rozdielom oproti modelu A je zvýšenie maximálneho počtu epoch na 300, aby mal model dostatočný priestor na trénovanie a možnosť dosiahnuť lepší stupeň generalizácie.



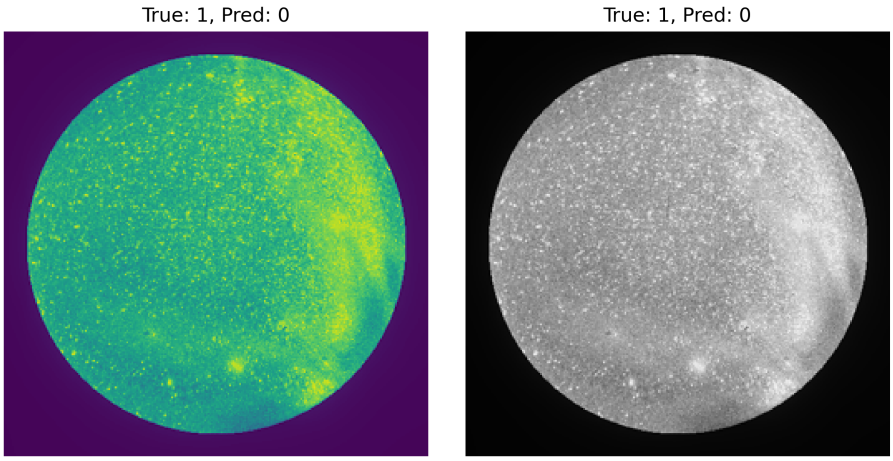
Obrázok 3 – 8 Matica zámen pre vyhodnotenie klasifikácie modelu A32,64

Layer (type)	Output Shape	Param #
conv2d (Conv2D)	(None, 254, 254, 32)	320
max_pooling2d (MaxPooling2D)	(None, 127, 127, 32)	0
dropout (Dropout)	(None, 127, 127, 32)	0
conv2d_1 (Conv2D)	(None, 125, 125, 64)	18496
max_pooling2d_1 (MaxPooling2D)	(None, 62, 62, 64)	0
dropout_1 (Dropout)	(None, 62, 62, 64)	0
conv2d_2 (Conv2D)	(None, 60, 60, 128)	73856
max_pooling2d_2 (MaxPooling2D)	(None, 30, 30, 128)	0
dropout_2 (Dropout)	(None, 30, 30, 128)	0
flatten (Flatten)	(None, 115200)	0
dense (Dense)	(None, 256)	29491456
dropout_3 (Dropout)	(None, 256)	0
dense_1 (Dense)	(None, 3)	771

Obrázok 3 – 9 Architektúra druhého modelu B32,64,128.

Trieda	Presnosť	Návratnosť	f1 skóre
0 - **cloudy**	0,98	1,00	0,99
1 - **partly**	0,93	0,82	0,87
2 - **clear**	0,87	0,95	0,91
makro priemer	0,93	0,92	0,92
vážený priemer	0,92	0,92	0,92

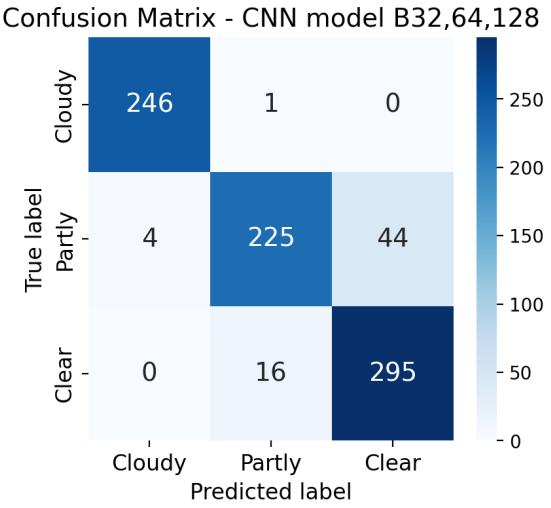
Tabuľka 3 – 5 Vyhodnotenie druhého modelu B32,64,128 na testovacích dátach.



Obrázok 3 – 10 Príklad nesprávnej klasifikácie obrázku modelom B32,64,128.

Trénovanie modelu trvalo o niečo dlhšie v porovnaní s prvým modelom. Najlepší model, ktorý sa uložil na základe najnižšej hodnoty validačnej chyby, bol dosiahnutý v epoche 219 s hodnotou *val_loss* 0,18.

Na záver môžeme povedať, že model B si vedie veľmi dobre, čo potvrdzujú aj výsledky uvedené v tabuľke 3 – 5. Jediný čiastočný problém sa vyskytuje pri triede **partly**, kde model nedokáže spoľahlivo zachytiť drobné oblačnosti na inak jasných obrázkoch nočnej oblohy. Príklad takejto nesprávnej klasifikácie je možné vidieť na obrázku 3 – 10. Výsledky testovania modelu B môžeme vidieť na obrázku 3 – 11.



Obrázok 3 – 11 Matica zámen pre vyhodnotenie klasifikácie modelu B32,64,128.

Model C32,64,128,256

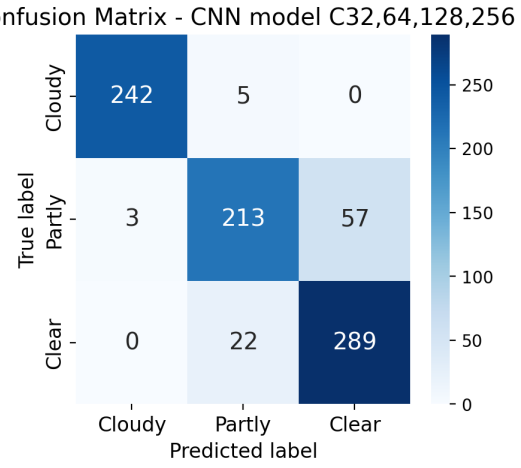
Tretí model bol opäť o niečo zložitejší ako model B, pričom jeho cieľom bolo otestovať, či ďalšie zvýšenie komplexity architektúry povedie k ešte lepším výsledkom. Do modelu sme pridali štvrtú konvolučnú vrstvu s 256 filtrami. Výsledný model, označený ako C32,64,128,256, tak obsahuje štyri konvolučné vrstvy, ktoré postupne využívajú 32, 64, 128 a 256 filtrov. Za každou konvolučnou vrstvou je aplikovaná vrstva `MaxPooling2D` a následne `Dropout`, pričom hodnoty `Dropout` sa zvyšujú od 0,2 po 0,5. Plne prepojené vrstvy zostávajú nezmenené oproti predchádzajúcemu modelu, teda jedna skrytá `Dense` vrstva s 256 neurónmi a výstupná `Dense` vrstva s tromi výstupmi pre jednotlivé triedy, zakončená `softmax` aktiváciou.

Trénovanie tretieho modelu trvalo kratšie ako v prípade modelu B. Model C dosiahol najnižšiu hodnotu *val_loss* už v epoche 134 s hodnotou 0,22. V porovnaní s modelom B model C dosahuje čiastočne horšie výsledky, čo je zrejmé aj z tabuľky vyhodnotenia 3 – 6. Z tejto tabuľky môžeme pozorovať, že hoci *precision* pre triedu 0 (**cloudy**) je nepatrne vyššia ako pri modeli B, ostatné metriky sú už porovnateľné alebo nižšie.

Model C je čiastočne horší oproti modelu B, nasvedčuje tomu aj matica zámen

Trieda	Presnosť	Návratnosť	f1 skóre
0 - **cloudy**	0,99	0,98	0,98
1 - **partly**	0,89	0,78	0,83
2 - **clear**	0,84	0,93	0,88
makro priemer	0,90	0,90	0,90
vážený priemer	0,90	0,90	0,89

Tabuľka 3 – 6 Vyhodnotenie tretieho modelu C32,64,128,256 na testovacích dátach.



Obrázok 3 – 12 Matica zámen pre vyhodnotenie klasifikácie modelu C32,64,128,256.

na Obrázku 3 – 12. Model C má väčšiu tendenciu pri snímkoch s triedou **partly** predikovať triedu **clear**.

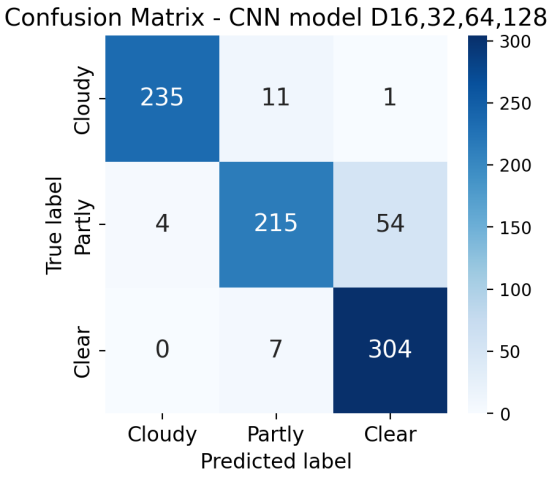
Model D16,32,64,128

Vzhľadom na to, že ďalšie zvyšovanie počtu vrstiev už neprinieslo zlepšenie, rozhodli sme sa pre štvrtý model aplikovať odlišnú stratégiu. Model D vychádzal z architektúry modelu C, avšak s polovičným počtom neurónov v každej konvolučnej vrstve: 16, 32, 64 a 128. Hodnoty atribútov `Dropout` ostali nezmenené v porovnaní s predchádzajúcim modelom. Cieľom zníženia počtu neurónov bolo zistiť, či je výhodnejšie mať menej vrstiev s vyšším počtom filtrov, alebo viac vrstiev s nižším počtom.

Model D podobne ako model C nedosiahol lepšie výsledky. Najlepší model bol

Trieda	Presnosť	Návratnosť	f1 skóre
0 - **cloudy**	0,98	0,95	0,97
1 - **partly**	0,92	0,79	0,85
2 - **clear**	0,85	0,98	0,91
makro priemer	0,92	0,91	0,91
vážený priemer	0,91	0,91	0,91

Tabuľka 3 – 7 Vyhodnotenie modelu D16,32,64,128 na testovacích dátach.



Obrázok 3 – 13 Matica zámen pre vyhodnotenie klasifikácie modelu D16,32,64,128.

dosiahnutý v epoche 123 s hodnotou *val_loss* 0,28 a úspešnosťou 0,91 na testovacích dátach. Môžeme teda usúdiť, že aj keď model D dosahuje dobrú úspešnosť, ako môžeme vidieť na Obrázku 3 – 13, model B je stále o niečo lepší. Z uvedeného obrázka zároveň môžeme vyčítať, že model D nesprávne klasifikoval jeden obrázok triedy **cloudy** ako **clear**, čo sa pri modeloch vyššej kvality zvyčajne nevyskytovalo. V porovnaní s modelom C model D ukazuje zlepšenie, najmä v správnej klasifikácii triedy **clear** a zároveň návratnosť triedy **clear** ktorú môžeme vidieť v Tabuľke 3 – 7 je 0,98, ale ako bolo povedané, model B stále dosahuje najlepšiu úroveň správnej klasifikácie.

Model E32,64,128

Keďže model B má zatiaľ najlepšiu architektúru, vytvorili sme ešte jeden model, model E, ktorý mal rovnakú architektúru ako model B, no zmenili sme hodnotu parametra `BATCH_SIZE` na 64, aby sme porovnali, či väčší objem trénovacích dát v jednom „Batchi“ pomôže modelu E prekonať model B.

Model E32,64,128 mal najnižšiu hodnotu *validation loss* 0,22, ktorú dosiahol v 191. epoche. Výsledky tohto modelu sú veľmi podobné výsledkom modelu B, ako môžeme vidieť na Obrázku 3 – 14. Úspešnosť predikcií na testovacích dátach bola 0,90. Všetky metriky vyhodnotenia modelu E je možné vidieť v Tabuľke 3 – 8. Zmena parametra `BATCH_SIZE` teda nemala výrazný vplyv na výkonnosť modelu a model B tak naďalej zostáva najlepším modelom spomedzi všetkých testovaných architektúr.

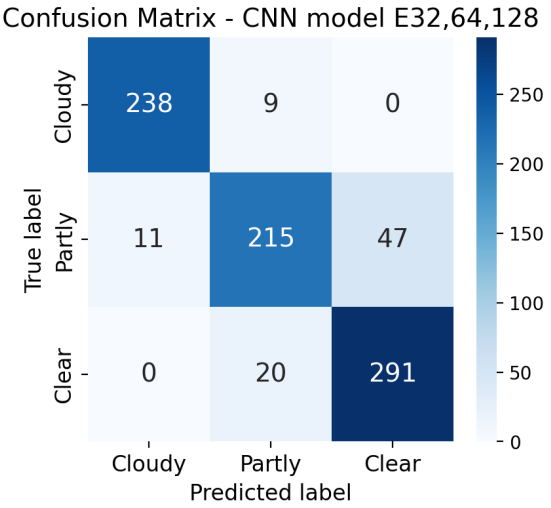
Trieda	**Presnosť**	**Návratnosť**	**f1 skóre**
0 - **cloudy**	0,96	0,96	0,96
1 - **partly**	0,88	0,79	0,83
2 - **clear**	0,86	0,94	0,90
makro priemer	0,90	0,90	0,90
vážený priemer	0,90	0,90	0,89

Tabuľka 3 – 8 Vyhodnotenie modelu E32,64,128 na testovacích dátach.

Model Z32,64,128

Pred finálnym vyhodnotením všetkých vytvorených modelov sme sa ešte pokúsili vytvoriť špeciálny typ konvolučného modelu, ktorý bol rozšírený o *attention* vrstvu. Attention mechanizmus je inšpirovaný spôsobom, akým ľudská pozornosť selektívne spracováva len niektoré časti vstupu. V prípade neurónových sietí umožňuje modelu zamerať sa na dôležité oblasti obrazu a priradiť im vyššiu váhu pri spracovaní, zatiaľ čo menej dôležité časti sú potlačené.

Model Z32,64,128 je založený na architektúre modelu B32,64,128, ktorá sa dote-



Obrázok 3 – 14 Matica zámen pre vyhodnotenie klasifikácie modelu E32,64,128.

raz ukázala ako najefektívnejšia. Do tejto architektúry sme pridali *attention* vrstvu, ktorá sa aplikuje na výstup poslednej konvolučnej vrstvy. Táto vrstva je realizovaná ako konvolučná vrstva s aktivačnou funkciou *sigmoid*, ktorej výstupom sú váhy pozornosti. Následne sa tieto váhy násobia so vstupom, čím sa zvýraznia relevantné črty v obraze (Jetley et al., 2018). Architektúru modelu Z je možné vidieť na Obrázku 3 – 15.

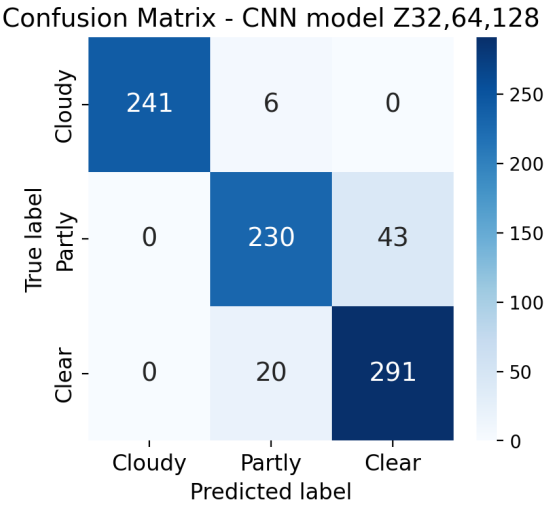
Posledný model Z dosiahol najnižšiu hodnotu *val_loss*, konkrétne 0,14 v epoche 214. Táto hodnota je najnižšia spomedzi všetkých modelov, pre porovnanie model B mal hodnotu *val_loss* 0,18. Napriek tomu, že model Z vykazuje lepšie výsledky na validačnej množine, pri testovacích dátach sú výsledky takmer totožné s modelom B, ako naznačuje aj tabuľka vyhodnotenia 3 – 9 a Obrázok 3 – 16. Môžeme teda konštatovať, že aj keď attention mechanizmus pomohol modelu lepšie sa prispôsobiť validačným dátam, na neznámych testovacích dátach nedošlo k výraznému zlepšeniu a úspešnosť ostáva rovnaká, teda 0,92.

Layer (type)	Output Shape	Param #	Connected to
input_2 (InputLayer)	[(None, 256, 256, 1)]	0	[]
conv2d_4 (Conv2D)	(None, 256, 256, 32)	320	['input_2[0][0]']
max_pooling2d_3 (MaxPooling2D)	(None, 128, 128, 32)	0	['conv2d_4[0][0]']
conv2d_5 (Conv2D)	(None, 128, 128, 64)	18496	['max_pooling2d_3[0][0]']
max_pooling2d_4 (MaxPooling2D)	(None, 64, 64, 64)	0	['conv2d_5[0][0]']
conv2d_6 (Conv2D)	(None, 64, 64, 128)	73856	['max_pooling2d_4[0][0]']
max_pooling2d_5 (MaxPooling2D)	(None, 32, 32, 128)	0	['conv2d_6[0][0]']
conv2d_7 (Conv2D)	(None, 32, 32, 128)	16512	['max_pooling2d_5[0][0]']
multiply_1 (Multiply)	(None, 32, 32, 128)	0	['max_pooling2d_5[0][0]', 'conv2d_7[0][0]']
flatten_1 (Flatten)	(None, 131072)	0	['multiply_1[0][0]']
dense_2 (Dense)	(None, 256)	33554688	['flatten_1[0][0]']
dropout_1 (Dropout)	(None, 256)	0	['dense_2[0][0]']
dense_3 (Dense)	(None, 3)	771	['dropout_1[0][0]']

Obrázok 3 – 15 Architektúra modelu Z32,64,128.

Trieda	**Presnosť**	**Návratnosť**	**f1 skóre**
0 - **cloudy**	1,00	0,98	0,99
1 - **partly**	0,90	0,84	0,87
2 - **clear**	0,87	0,94	0,90
makro priemer	0,92	0,92	0,92
vážený priemer	0,92	0,92	0,92

Tabuľka 3 – 9 Vyhodnotenie posledného modelu Z32,64,128 na testovacích dátach.



Obrázok 3 – 16 Matica zámen pre vyhodnotenie klasifikácie modelu s attention vrstvou Z32,64,128.

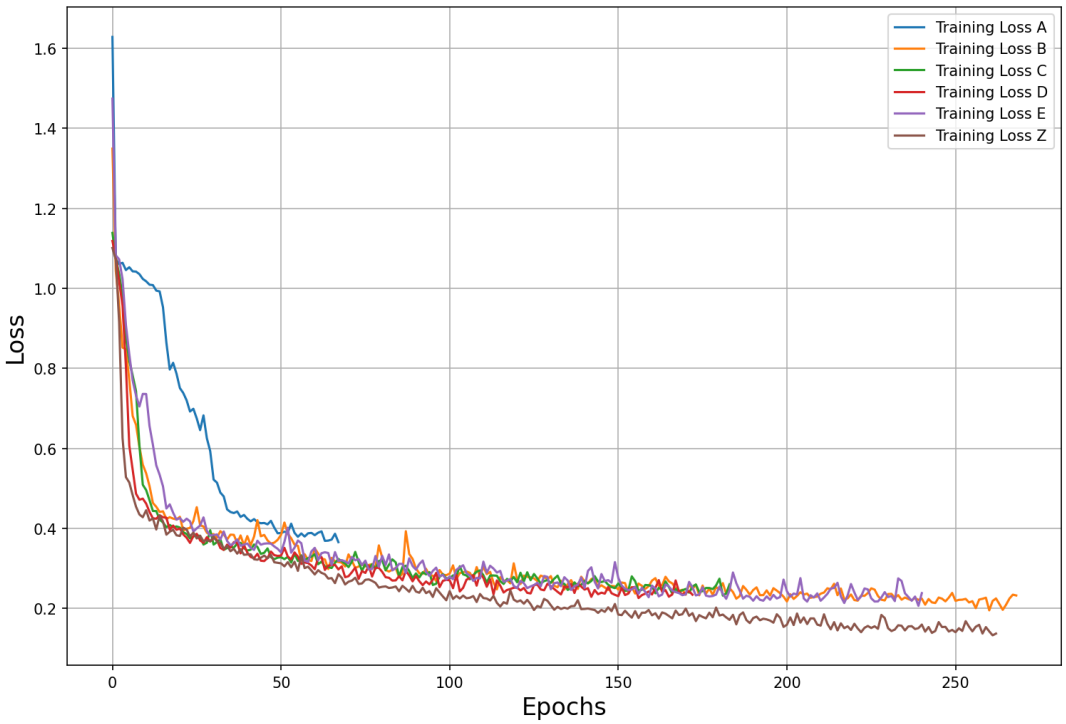
model	úspešnosť klasifikácie	presnosť	návratnosť	F1 skóre
A	0,80	0,83	0,80	0,79
B	0,92	0,92	0,92	0,92
C	0,90	0,90	0,90	0,90
D	0,91	0,91	0,91	0,91
E	0,90	0,90	0,90	0,89
Z	0,92	0,92	0,92	0,92

Tabuľka 3 – 10 Vyhodnocovacia tabuľka všetkých konvolučných modelov.

3.5 Vyhodnotenie

V tejto kapitole sa pozrieme na celkové zhodnotenie všetkých vytvorených modelov. V Tabuľke 3 – 10 sú prehľadne uvedené jednotlivé modely spolu s ich úspešnosťou klasifikácie, váhovo spriemerovanou presnosťou, návratnosťou a F1 skóre na testovacích dátach. Okrem najjednoduchšieho modelu A, ktorý dosiahol úspešnosť 0,80, všetky ostatné modely prekonali hranicu 0,90, čím výrazne prevýšili výkonnosť referenčných modelov z úvodnej časti práce, ako napríklad Random Forest.

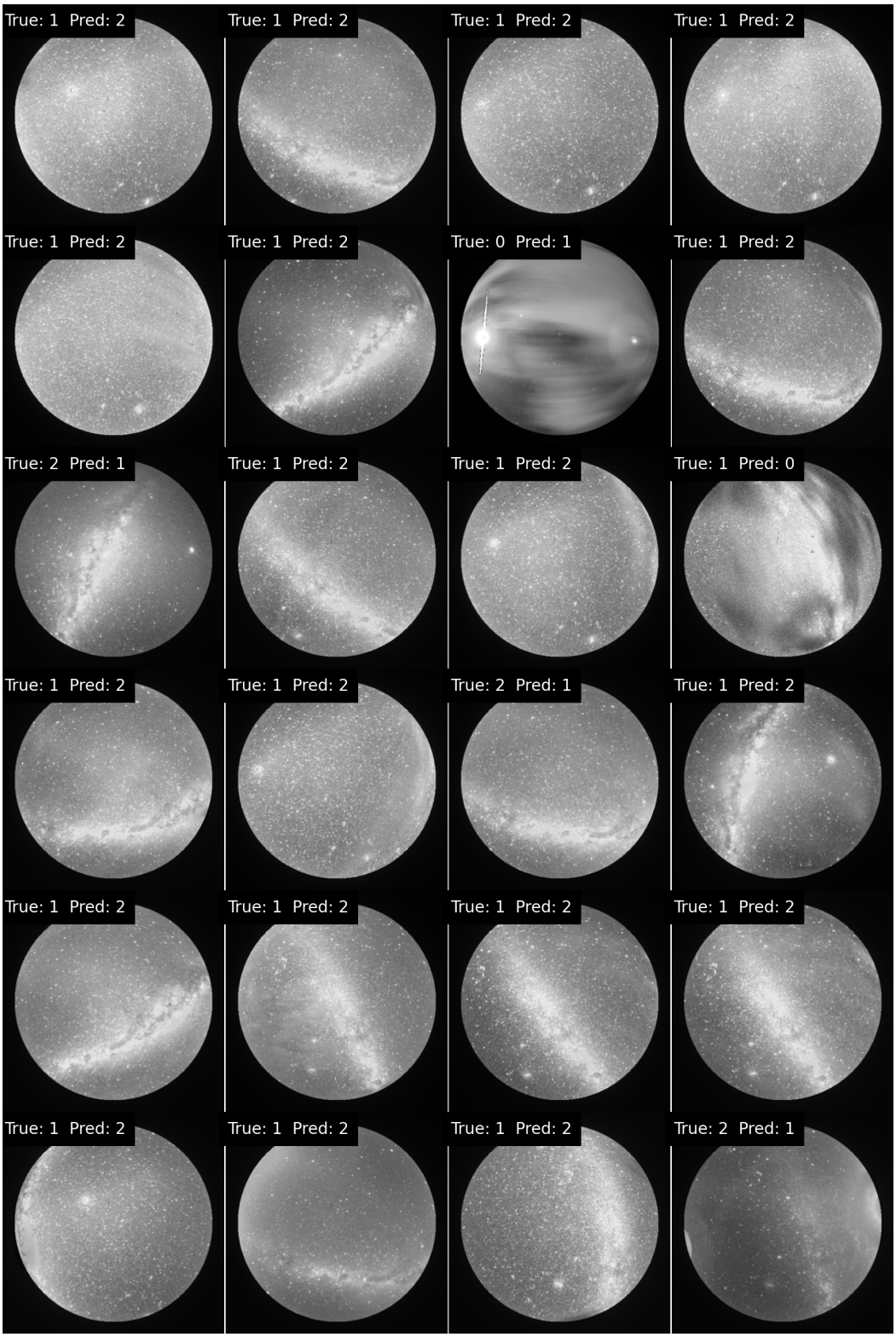
Ako najlepšia architektúra sa ukázala architektúra modelu B32,64,128, ktorá



Obrázok 3 – 17 Priebeh hodnôt chybových funkcií počas trénovania modelov A,B,C,D,E a Z

pozostáva z troch konvolučných vrstiev s počtom filtrov 32, 64 a 128. Tento model dosahoval konzistentne dobré výsledky nielen na validačných, ale aj na testovacích dátach. Zaujímavé je, že aj model Z, ktorý obsahoval dodatočnú *attention* vrstvu, mal výsledky prakticky zhodné s modelom B, a teda môžeme usúdiť, že v tomto prípade attention mechanizmus nezohral výraznú úlohu pri zlepšení klasifikácie, a to napriek tomu, že hodnota chybovej funkcie počas trénovania modelu Z bola viditeľne nižšia v porovnaní s ostatnými modelmi. Porovnanie hodnôt chybových funkcií všetkých modelov môžeme vidieť na Obrázku 3 – 17.

Treba však zdôrazniť, že výkonnosť týchto modelov je úzko prepojená s kvalitou vstupných dát. Platí, že čím viac dát máme na trénovanie, tým robustnejší môže byť výsledný model. Okrem kvantity však hrá dôležitú úlohu aj samotné anotovanie. V prípade nášho datasetu bolo občas nejednoznačné, kedy ešte hovoriť o kategórii **partly cloudy** a kedy už o jasnej nočnej oblohe, ako môžeme vidieť na Obrázku 3 – 18. Podobne mohla byť problematická aj klasifikácia obrazov s drobnými oblakmi,



Obrázok 3 – 18 Ukážka nesprávne klasifikovaných obrázkov nočnej oblohy modelom B.

ktoré môžu byť ľahko prehliadnuté.

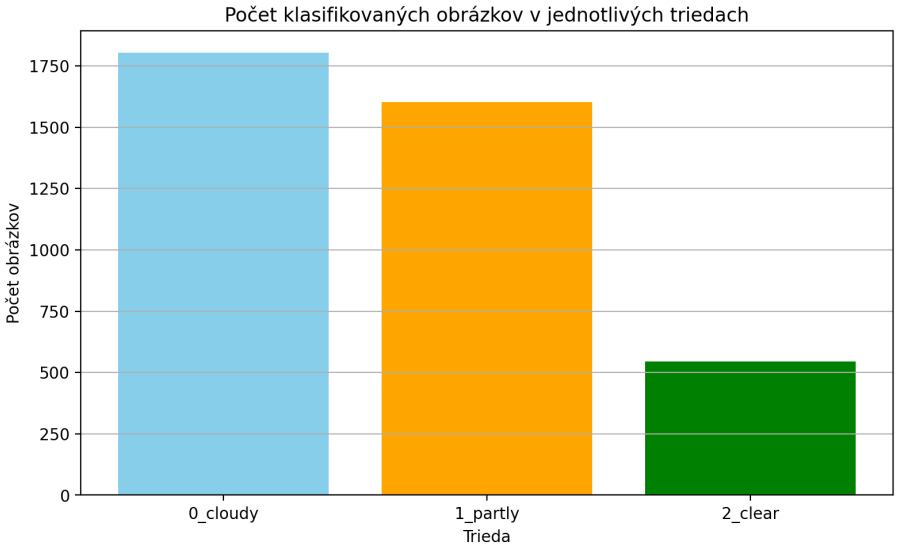
Na záver môžeme konštatovať, že konvolučné neurónové siete sa ukázali ako veľmi efektívne riešenie pre úlohu klasifikácie nočných oblohových snímok. Ich flexibilita a schopnosť učiť sa črty priamo z dát z nich robí vhodný nástroj pre vizuálne klasifikačné úlohy.

3.6 Nasadenie

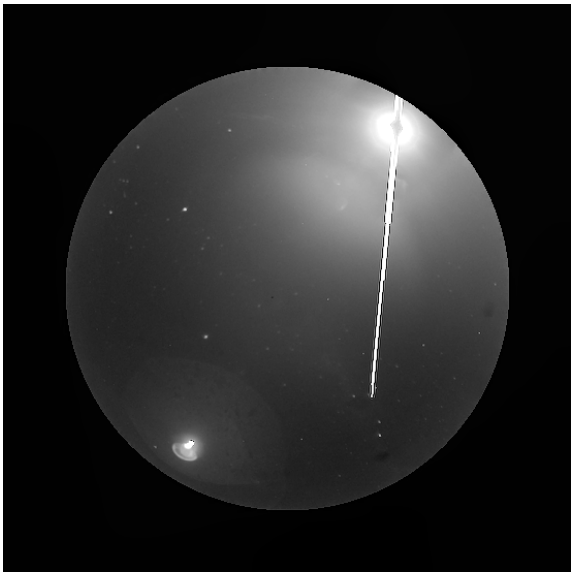
Na základe dosiahnutých výsledkov bol ako najvhodnejší model pre využitie v praxi vybraný model s architektúrou B32,64,128, ktorý dosiahol najlepšiu výkonnosť počas testovania. Tento model bol následne aplikovaný na nové, doposiaľ nevidené snímky z roku 2019, pochádzajúce zo stanice Mt. John na Novom Zélande, konkrétne z filtra 572,7 nm.

Po nasadení modelu B32,64,128 na tieto nové dáta môžeme konštatovať, že model si viedol veľmi dobre a výsledky zodpovedajú očakávaniam. Podrobné rozdelenie obrázkov podľa klasifikačných tried je znázornené na Obrázku 3 – 19. Okolo 1 800 obrázkov bolo klasifikovaných do triedy **cloudy**, okolo 1 600 do triedy **partly_cloudy** a niečo vyše 500 bolo klasifikovaných do triedy **clear**. Väčšina snímok bola správne zaradená do príslušných kategórií, avšak niektoré snímky vyvolávajú určité otázky. Napríklad Obrázok 3 – 20 zobrazuje scénu bez oblačnosti, no zároveň obsahuje jasný mesiac, čo môže ovplyvniť kvalitu klasifikácie, keďže hviezdy sú na snímke takmer neviditeľné. Takéto prípady ukazujú, že hoci model funguje spoľahlivo vo väčšine prípadov, stále existujú situácie, ktoré môžu byť predmetom ďalšieho zlepšenia alebo doplnenia o ďalšie kritériá.

Aj keď samotná implementácia a integrácia modelu do reálnej prevádzky nie je priamo súčasťou tejto bakalárskej práce, získané výsledky predstavujú významný príspevok pre ďalšie výskumy a experimenty. Môžu slúžiť ako základ pre nadväzujúce výskumné aktivity, ako je napríklad automatizované filtrovanie veľkých dátových súborov a následné podrobnejšie štúdium špecifických javov v atmosfére, ako sú



Obrázok 3 – 19 Rozdelenie nových snímok zo stanice Mt John do tried modelom B32,64,128.



Obrázok 3 – 20 Obrázok nočnej oblohy zo stanice Mt John klasifikovaný do triedy **clear**.

napríklad plazmové bubliny. Model a jeho výstupy tak ponúkajú dôležitý nástroj pre ďalšiu analýzu snímok zo zariadenia ASI.

4 Záver

Cieľom tejto práce bolo navrhnúť a vyhodnotiť modely hlbokého učenia na klasifikáciu snímok oblohy do tried podľa oblačnosti, čo predstavuje dôležitý krok smerom k automatizovanému predspracovaniu dát zo zariadení kamier ASI. Na úvod sme poskytli teoretický prehľad problematiky neurónových sietí so zameraním na konvolučné neurónové siete, ktoré sú v súčasnosti štandardom pri úlohách spracovania obrazových dát. V tejto časti sme zároveň zahrnuli stručnú analýzu súčasného stavu a existujúcich prístupov v danej oblasti.

V praktickej časti sme najskôr vytvorili niekoľko referenčných modelov, konkrétne rozhodovací strom, model náhodného lesa a logistickú regresiu, ktoré slúžili ako referenčné body pre porovnanie s pokročilejšími prístupmi. Následne sme pripravili obrazové dáta (orez a normalizácia) a navrhli šesť rôznych architektúr konvolučných neurónových sietí. Ich výkonnosť sme porovnali na testovacích dátach, pričom najlepšie výsledky dosiahol model B32,64,128 a model Z32,64,128 s *attention* vrstvou.

Tento model sme potom aplikovali na nové, doposiaľ nevidené snímky zo stanice Mt. John z roku 2019 vo filtri 572,7 nm, čím sme overili jeho použiteľnosť na reálne dáta v praxi. Snímky boli automaticky roztriedené do priečinkov podľa predikovaných tried, čo môže výrazne zjednodušiť následné analýzy.

Na záver možno skonštatovať, že konvolučné neurónové siete sa ukázali ako užitočný a efektívny nástroj pre klasifikáciu snímok z kamier ASI. Aj keď cieľom práce nebolo vytvoriť riadne nasadený systém, natrénované modely majú praktický potenciál a môžu byť priamo využité ako súčasť automatizovaného spracovania snímok alebo ako podklad pre ďalší výskum, napríklad pri analýze javov, ako sú plazmové bubliny.

Pri písaní niektorých častí tejto práce som využil generatívne jazykové modely ako podporu pri tvorbe textu, úprave štýlu a gramatiky. Odborný obsah, interpretácia a konečná podoba práce sú však výsledkom mojej vlastnej práce.

Literatúra

Bhattacharyya, A. (2022). Equatorial plasma bubbles: A review, *Atmosphere* **13**(10): 1637.

Cazorla Cabrera, A., Husillos, C., Antón, M., Alados Arboledas, L. et al. (2015). Multi-exposure adaptive threshold technique for cloud detection with sky imagers.

Chollet, F. and Chollet, F. (2021). *Deep learning with Python*, Simon and Schuster.

Duchi, J., Hazan, E. and Singer, Y. (2011). Adaptive subgradient methods for online learning and stochastic optimization., *Journal of machine learning research* **12**(7).

Goodfellow, I., Bengio, Y. and Courville, A. (2016). *Deep Learning*, MIT Press. `http://www.deeplearningbook.org`.

Jetley, S., Lord, N. A., Lee, N. and Torr, P. H. (2018). Learn to pay attention, *arXiv preprint arXiv:1804.02391* .

Kingma, D. P. and Ba, J. (2014). Adam: A method for stochastic optimization, *arXiv preprint arXiv:1412.6980* .

Krstinić, D., Braović, M., Šerić, L. and Božić-Štulić, D. (2020). Multi-label classifier performance evaluation with confusion matrix, *Computer Science & Information Technology* **1**: 1–14.

Mackovjak, Š., Martinis, C., Wroten, J., Baumgardner, J. and Mendillo, M. (2025). Synergy of traditional techniques and convolutional neural networks for classification of cloudless conditions by all-sky imagers, *Publications of the Astronomical Society of the Pacific* **137**(4): 045004.

Martinis, C., Baumgardner, J., Wroten, J. and Mendillo, M. (2018). All-sky-imaging capabilities for ionospheric space weather research using geomagnetic conjugate point observing sites, *Advances in Space Research* **61**(7): 1636–1651.

Mommert, M. (2020). Cloud identification from all-sky camera data with machine learning, *The Astronomical Journal* **159**(4): 178.

O'shea, K. and Nash, R. (2015). An introduction to convolutional neural networks, *arXiv preprint arXiv:1511.08458* .

Sharma, S., Sharma, S. and Athaiya, A. (2017). Activation functions in neural networks, *Towards Data Sci* **6**(12): 310–316.

Solomon, S. C. and Abreu, V. J. (1989). The 630 nm dayglow, *Journal of Geophysical Research: Space Physics* **94**(A6): 6817–6824.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I. and Salakhutdinov, R. (2014). Dropout: a simple way to prevent neural networks from overfitting, *The journal of machine learning research* **15**(1): 1929–1958.

Tieleman, T. (2012). Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude, *COURSERA: Neural networks for machine learning* **4**(2): 26.

Yacoub, M., Abdelwahab, M., Shiokawa, K. and Mahrous, A. (2025). Automatic detection of equatorial plasma bubbles in airglow images using two-dimensional principal component analysis and explainable artificial intelligence, *Machine Learning and Knowledge Extraction* **7**(1): 26.

Zoznam príloh

Príloha A CD médium – záverečná práca v elektronickej podobe, prílohy v elektronickej podobe. [2]

Príloha B Systémová a používateľská príručka

[2]Prílohy dostupné elektronicky na: `https://github.com/kkuichi/lk677jt`