

Technická univerzita v Košiciach
Fakulta elektrotechniky a informatiky

Automatická detekcia meteorov na
rádiových spektrogramoch

Bakalárska práca

2022

Kamila Jenčíková

Technická univerzita v Košiciach
Fakulta elektrotechniky a informatiky

Automatická detekcia meteorov na rádiových spektrogramoch

Bakalárska práca

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

Košice 2022

Kamila Jenčíková

Abstrakt v SJ

V dnešnej dobe už existujú meracie aparatúry, ktoré zameriavajú meteory na obrovskú vzdialenosť. Meraniami sa získavajú dáta, ktorých analýza poskytuje veľmi dôležité a zaujímavé informácie. Dát je ale veľké množstvo a na ich spracovanie častokrát nepostačujú len ľudské sily, preto je dôležité využitie techník strojového učenia. Cieľom tejto bakalárskej práce je vytvorenie modelu neurónovej siete, ktorý na základe učenia, tréningu a testovania dokáže detekovať a následne klasifikovať existenciu meteorických javov na oblohe.

Klíčové slová

detekcia meteorov, bakalárska práca, rádiový spektrogram, ozvena, neurónová sieť, headecho, anotovanie

Abstrakt v AJ

Nowadays, there are already measuring devices that target meteors at a huge distance. The measurements obtain data, the analysis of which provides a lot of important and interesting information. However, there is a large amount of data and often only human resource is not enough to process them, so it is important to use machine learning techniques. The aim of this bachelor thesis is to create a neural network model that based on learning, training and testing can detect and then classify the existence of meteoric phenomena in the sky.

Klíčové slová v AJ

meteor detection, bachelor thesis, radio spectrogram, echo, neural network, headecho, annotation

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:

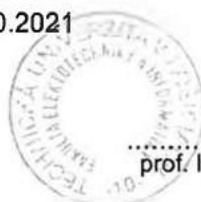
Automatická detekcia meteorov na rádiových spektrogramoch
Automatic detection of meteors in radio spectrograms

Študent: **Kamila Jenčíková**
Školiteľ: **doc. Ing. Peter Butka, PhD.**
Školiace pracovisko: **Katedra kybernetiky a umelej inteligencie**
Konzultant práce: **Ing. Viera Maslej Krešňáková, PhD., 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 vhodné metódy pre automatickú detekciu meteorov v rádiových spektrogramoch.
2. Analyzovať a predspracovať dostupnú dátovú množinu rádiových spektrogramov z meteorického radaru.
3. Pripraviť anotačný projekt a získať anotácie pre detekciu meteorov v rádiových spektrogramoch.
4. Navrhnuť, realizovať a vyhodnotiť model hlbokého učenia pre detekciu meteorov v danej množine dát.
5. Vypracovať dokumentáciu podľa pokynov katedry a vedúceho práce.

Jazyk, v ktorom sa práca vypracuje: slovenský
Termín pre odovzdanie práce: 27.05.2022
Dátum zadania bakalárskej práce: 29.10.2021



prof. Ing. Liberios Vokorokos, PhD.
dekan fakulty

Čestné vyhlásenie

Vyhlasujem, že som bakalársku prácu vypracovala samostatne s použitím uvedenej odbornej literatúry.

Košice 27. 5. 2022

.....

Vlastnoručný podpis

Podakovanie

Touto cestou by som sa chcela poďakovať všetkým, vďaka ktorým mohla táto práca vzniknúť. V prvom rade doc. Ing. Petrovi Butkovi, PhD., ktorý mi stále poskytoval užitočné informácie a dôležité pripomienky.

Ďalšia vďaka patrí konzultantovi RNDr. Šimonovi Mackovjakovi, PhD., vďaka ktorému som získala mnohé potrebné a veľmi zaujímavé informácie k doméne, a ktorý mi bol stále ochotný poradiť ako pokračovať v ďalších krokoch.

Veľká vďaka patrí mojej konzultantke Ing. Vieri Maslej Krešňákovej, PhD. za všetok čas, ktorý obetovala pre to, aby moja práca išla stále správnym smerom a za to, že mi bola k dispozícii stále, keď to bolo potrebné.

V neposlednom rade sa chcem poďakovať Ing. Pavlovi Zigovi, PhD., ktorý nám poskytol potrebné dáta a dôležité informácie na uskutočnenie tejto spolupráce.

Taktiež patrí veľká vďaka študentom Technickej univerzity v Košiciach a mojim priateľom, ktorí mi boli ochotní pomáhať s prípravou dát.

Obsah

Úvod	1
1 Úvod do neurónových sietí	3
1.1 Model neurónovej siete	3
1.2 Strojové učenie	3
1.2.1 Hlboké učenie	5
1.2.2 Štandardné metódy strojového učenia	9
1.3 YOLO	11
1.3.1 YOLOv5	13
2 Detekcia meteorov na rádiových spektrogramoch	16
2.1 Rozdelenie meteorov	16
2.1.1 Nízkohustotné – <i>Underdense</i>	17
2.1.2 Prehustené – <i>Overdense</i>	18
2.1.3 Hlava ozveny – <i>Headecho</i>	19
2.2 Analýza súčasného stavu	20
3 Automatická detekcia meteorov na rádiových spektrogramoch pomocou metód hlbokého učenia	23
3.1 Pochopenie úlohy	23
3.2 Pochopenie dát	24
3.3 Príprava dát v prostredí Zooniverse	26
3.3.1 Pracovné prostredie projektu	28
3.3.2 Zooniverse z pohľadu anotátora	30
3.4 Extrakcia anotovaných dát z prostredia Zooniverse	31
3.5 Príprava dát na modelovanie	33
3.6 Modelovanie	36
3.7 Vyhodnotenie riešenia	40

4	Záver	46
	Zoznam príloh	51

Zoznam obrázkov

1–1	Vennov diagram. Zdroj: (Ko et al., 2022)	4
1–2	Architektúra neurónovej siete. Zdroj: (Roopak et al., 2019)	7
1–3	Príklad klasifikácie obrázkov. Zdroj: (Albawi et al., 2017)	8
1–4	Sieť RNN. Zdroj: (Beysolow et al., 2017)	9
1–5	Príklad rozhodovacieho stromu. Zdroj: vlastné	10
1–6	Porovnanie YOLO a Fast R-CNN. Zdroj: (Wang et al., 2018)	12
1–7	Architektúra YOLO V4. Zdroj: (Aly et al., 2021)	13
1–8	Architektúra YOLO V5. Zdroj: (Xu et al., 2021)	15
2–1	Signály z rádiového vysieláča (angl. <i>transmitter</i>) sú do prijímača (angl. <i>receiver</i>) rozptýlené stopou meteoru, čo umožňuje detekciu meteoru. Zdroj: (Wislez, 1996)	16
2–2	Odraz rádiových vĺn na stopách meteorov je zrkadlový. Zdroj: (Wislez, 1996)	17
2–3	Rozloženie hustoty iónov v stope meteoru. Zdroj: (Wislez, 1996)	17
2–4	Typický meteor s nízkou hustotou, pozorovaný zariadením RAM-SES. Zdroj: (Wislez, 1996)	18
2–5	Typický meteor s prehustením, pozorovaný zariadením RAMSES. Zdroj: (Wislez, 1996)	19
2–6	Porovnanie meteoru bez <i>headecha</i> s meteorom s <i>headechom</i> . Zdroj: AGO Modra	20
2–7	Porovnanie snímky videa zachyteného meteoru a jeho zobrazenia prostredníctvom radaru GRAVES. Zdroj: (Fleet, 2015)	21
2–8	Príklady chybného detekovania meteorov GRAVES radaru. Zdroj: AGO Modra	21
3–1	Príklad druhov vyskytujúcich sa falošných javov. Zdroj: P.Zigo, AGO Modra	25
3–2	Príklad druhov <i>headecha</i> . Zdroj: P.Zigo, AGO Modra	25

3-3	Druhy krátkych ozvien U. Zdroj: P.Zigo, AGO Modra	26
3-4	Druhy dlhých ozvien O. Zdroj: P.Zigo, AGO Modra	26
3-5	Lišta prostredia Zooniverse pri vytváraní a úprave projektu	28
3-6	Možnosti volby úlohy v prostredí Zooniverse. Zdroj: vlastné	29
3-7	Prostredie Zooniverse viditeľné pre anotátorov. Zdroj: vlastné	30
3-8	Príklad časti .csv výstupu z prostredia Zooniverse. Zdroj: vlastné	31
3-9	Príklad časti vytvorenej výstupnej tabuľky. Zdroj: vlastné	34
3-10	Príklad textového súboru pripraveného z výstupných dát. Zdroj: vlastné	35
3-11	Príklad kódu tréovania neurónovej siete. Zdroj: vlastné	37
3-12	Zobrazenie yaml súboru s cestami k množinám. Zdroj: vlastné	37
3-13	Zobrazenie kódu pre testovanie dát. Zdroj: vlastné	39
3-14	Časť finálnej výstupnej tabuľky po modelovaní. Zdroj: vlastné	45
3-15	Príklad vyhodnotenia modelom. Zdroj: vlastné	45

Zoznam tabuliek

3-1	Tabuľka s rozdeleným počtom jednotlivých druhov ošetrovaných anotácií	34
3-2	Kontigenčná tabuľka klasifikácie	41
3-3	Sumarizácia validácie modelu	42

Zoznam symbolov a skratiek

AI Artificial Intelligence

ANN Artificial Neural Network

AP Average Precision

CNN Convolutional Neural Network

Concat Concatenate Function

Conv Convolutional Layer

CSP Cross Stage Partial Network

DL Deep Learning

FN False Negative

FNN Feedforward Neural Network

FP False Positive

FPN Feature Pyramid Network

GBM Gradient Boosting Machine

GRAVES Grand Réseau Adapté à la Veille Spatiale

ID Identification

IoU Intersection Over Union

IP Internet Protocol

KPI Key Performance Indicator

mAP mean Average Precision

ML Machine Learning

MLP Multilayer Perceptron

PANet Path Aggregation Network for Instance Segmentation

R-CNN Region Based Convolutional Neural Networks

RNN Recurrent Neural Networks

RMOB Radio Meteor Observing Bulletin

SLP Singlelayer Perceptron

SPP Spatial Pyramid Pooling

SVM Support Vector Machine

TN True Negative

TP True Positive

YOLO You Only Look Once

Úvod

Predložená bakalárska práca je zameraná na detekciu meteorov na rádiových spektrogramoch. Pozostáva z dvoch hlavných častí, a to zo získania dát prostredníctvom anotovania a ich následné využitie pri práci s detekčnou neurónovou sieťou YOLO (angl. *You Only Look Once*). Na základe toho sa pomocou detekčnej siete YOLO na tréovanie hlbokoj a neurónovej siete vytvorí nový model, ktorý vďaka automatickej detekcii objektov dokáže zachytiť sledované javy, v tomto prípade meteory. Tento model dokáže nie len detekovať, ale dokonca klasifikovať hľadané objekty, ktoré sú rozdelené do dvoch základných skupín.

Detekovanie meteorov je veľmi dôležité, pretože meteory stále nie sú úplne prebádané. Preto sme sa rozhodli zamerať na oblasť vesmíru, konkrétne tému meteorov. Keďže je získaných dát veľa a ľudské sily na to častokrát nestačia, je veľmi dôležité hľadať možnosti, ako zachytené dáta rýchlo spracovať.

Je dôležité uvedomiť si, že rozoznanie ozvien je výsledkom **hlbokého učenia** neurónovej siete. Naším poskytnutím informácií sa dokáže naučiť, ako dané javy vyzerajú a tým ich rozoznať. Táto metóda je vo svete stále viac známa. Skvelým faktom je, že učenie neurónovej siete je možné vykonať na rôznych javoch. Príkladom môže byť rozpoznávanie ľudskej tváre pri zabezpečení telefónov. Mnohé automobilové firmy tieto výdobytky zaradili napríklad k vytvoreniu bezpečnejšej jazdy prostredníctvom zamerania, zdetekovania a rozpoznanie pohybujúcej sa zvery v okolí ciest. Cieľový program dokáže teda rozoznať nie len druh ozveny, ale aj jej vlastnosť.

Všetky vyššie uvedené kroky sú vykonané na zozbieraných dátach. Vďaka využitiu **crowdsourcingového projektu**, na ktorom sa podieľali študenti Technickej univerzity v Košiciach, sa dané obrázky podarilo zanotovať a tým ich roztriediť do spomínaných skupín. Stiahnuté dáta vo formáte .csv z prostredia Zooniverse sú ďalej spracované v programovacom jazyku Python, v ktorom sa zozbierané dáta očistia od nepotrebných informácií a upravia na ďalšie spracovanie v prostredí YOLO.

V prvej kapitole tejto práce je predstavená téma neurónových sietí a jej modely.

V jej druhej časti je bližšie popísané strojové učenia, v tretej je priblížená funkcia YOLO.

Druhá kapitola obsahuje bližšie informácie o prostredí, ktoré sa použilo na vytvorenie spomínaného detekčného modelu. Venuje sa jednotlivým popisom ozvien, ich rozdielom, tomu, ako vznikajú, alebo tomu, čo môžu obsahovať. Jej druhá časť je venovaná priblíženiu súčasného stavu. Ten je znázornený najmä na základe už existujúceho riešenia, ktorým je detekčný model GRAVES. Zameriava sa najmä na jeho nevýhody a tým predostrieva čitateľovi pohľad na dôležitosť nájdenia a vytvorenia lepšieho riešenia.

V tretej kapitole je zobrazené celkové pochopenie dát, cieľa, ich príprava a následné výstupy. Predstavujú sa v nej používané nástroje na úpravu dát, problémy, s ktorými sme sa pri ich príprave mohli stretnúť, prostriedky, alebo ľudia, ktorí boli potrební na to, aby ich anotovanie prebiehalo rýchlejšie a efektívnejšie. Záver tejto kapitoly je venovaný informáciám o modelovaní, teda tvorbe modelu prostredníctvom YOLOv5 a vyhodnoteniam jeho výsledkov. Taktiež je popísaná celá cesta tréňovaním, testovaním a vyhodnocovaním a predostiera bližšie informácie o tom, ako jednotlivé časti modelu fungujú.

1 Úvod do neurónových sietí

1.1 Model neurónovej siete

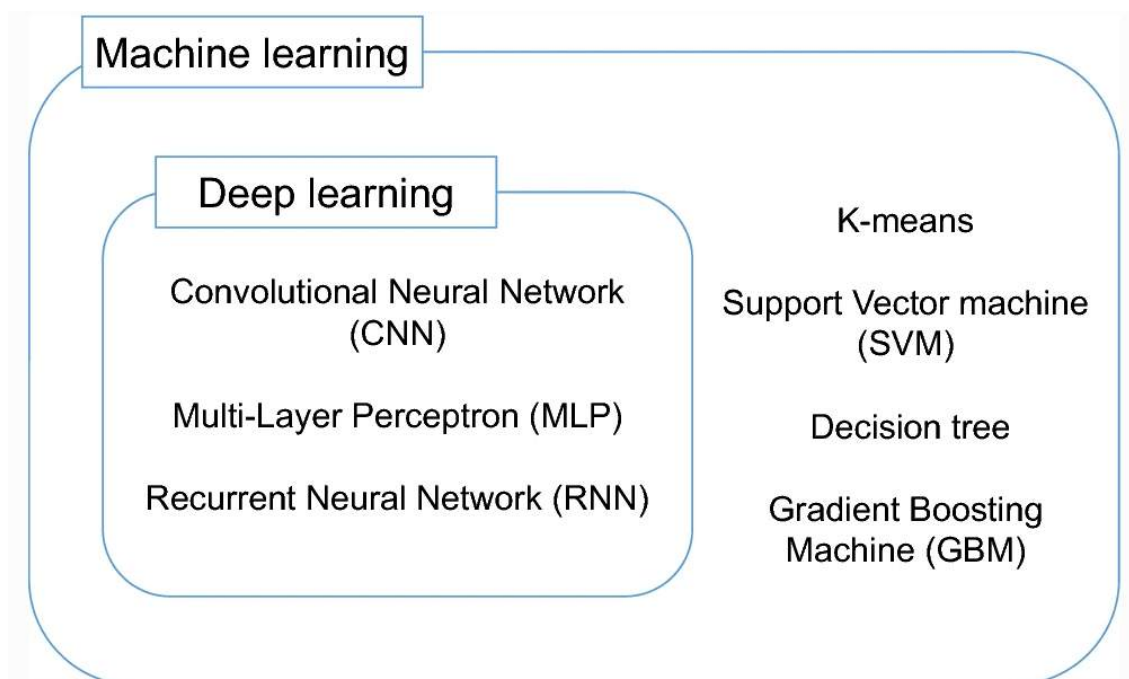
Umelá inteligencia (angl. *artificial intelligence*, skr. AI) sa začala vyvíjať rýchlym tempom a vďaka tomu dosahuje veľké pokroky v mnohých oblastiach, ako je oblasť medicíny, vedy, techniky, vzdelávania, polície, výroby, či marketingu. To zabezpečilo fakt, že v mnohých smeroch už prekonala aj sily samotných odborníkov v jednotlivých oblastiach. Dôležitým odvetvím AI je **strojové učenie** (angl. *machine learning*, skr. ML), ktoré je reprezentáciou schopnosti stroja identifikovať vzťahy medzi dátami bez explicitných kritérií. Ich identifikáciu je možné zlepšovať zvyšovaním množstva dát a skúseností, čo následne umožňuje jednotlivým algoritmom vytvárať zložitejšie modelovacie vzťahy. To je napríklad problémom pre iné, štandardné metódy.

Na základe vytvorenia modelu, ktorý je trénovaný na veľkom množstve nami označených dát, využívame spomínaný algoritmus strojového učenia. Ten prispieva k tomu, že sa model na základe poznania už známych dát učí rozoznávať tie neznáme. Klasifikovať sa môžu rôzne oblasti, táto práca je ale zameraná na klasifikáciu obrazu. Na Obrázku 1–1 je znázornený Vennov diagram, ktorý predstavuje typické príklady strojového učenia a hlbokého učenia (angl. *deep learning*, skr. DL) (Ko et al., 2022). Jeho časti sú bližšie vysvetlené nižšie v tejto kapitole.

1.2 Strojové učenie

Strojové učenie sa venuje otázke automatického zlepšovania počítačov prostredníctvom skúseností. Táto oblasť je zaznamenaná ako najrýchlejšie sa vyvíjajúca v dnešnej dobe a jej posledný pokrok bol na základe vývoja nových algoritmov a teórie učenia sa (Jordan and Mitchell, 2015).

Je dôležité nepredstavovať si proces učenia sa zo vstupných údajov ako fixne naprogramovaný, pretože ide o algoritmy nazývané ako *soft coded*, teda mäkko za-



Obrázok 1 – 1 Vennov diagram. Zdroj: (Ko et al., 2022)

kódované. Je to z toho dôvodu, že sú nútené stále meniť, resp. prispôbovať svoju architektúru opakovaní, teda skúsenosti, na základe ktorej sa zlepšujú a tým približujú bližšie k presným výsledkom (El Naqa and Murphy, 2015). Tento proces je známy pod názvom tréning. Tréning obsahuje množinu dát, ktoré obsahujú informácie o vstupe spolu s informáciami o požadovanom výstupe. Na základe toho sa model naučí ako neskôr rozoznať na neznámych dátach hľadané javy. Tento proces strojového učenia predstavuje učenie (angl. *learning*) (Roopak et al., 2019). Navyše je daný proces zvaný aj ako celoživotný, pretože sa dokáže zdokonaľovať na základe svojich chýb.

Ako bolo spomínané už v úvode tejto kapitoly, daný algoritmus má mnoho výhod. Nie len, že úspešne nahrádza namáhavé ľudské úsilie použité napríklad pri dlhých a opakujúcich sa algoritmoch, ale využíva aj svoju vysokú citlivosť rozoznávania. To napríklad vtedy, keď dokáže zo vstupných dát zachytiť jemnejšie a zložitejšie príznaky, na základe ktorých neskôr rozozná aj také javy, ktoré by ľudské oko nemuselo vôbec zaznamenať.

Z historického hľadiska siahajú jeho začiatky až do sedemnásteho storočia, kde si ich spájame s Blaiseom Pascalom a Gottfriedom Wilhelmom Leibnizom. Neskôr v roku 1959 Arthur Samuel prvý krát použil výraz **strojové učenie**, keď naučil počítač hrať dámu. V roku 1958 dosiahol Frank Rosenblatt vývoj perceptrónu ako jednej z prvých pomyselných architektúr neurónových sietí. Tak bol vďaka Paulovi Werbosovi v roku 1975 dosiahnutý prelom v oblasti perceptrónu, ktorý nastal na základe vývoja viacvrstvového perceptrónu (angl. *multilayer perceptron*, skr. MLP). Ďalší pokrok bol zaznamenaný v roku 1986, kedy John Ross Quinlan vyvinul rozhodovacie stromy (angl. *decision trees*) a dodnes sa celá oblasť stále viac a viac vyvíja a zlepšuje (El Naqa and Murphy, 2015).

Neurónové siete dokážu už dnes pracovať s akýmikoľvek spočítateľnými funkciami. Ich najväčšie využitie je práve pre klasifikáciu, funkčnú aproximáciu, či mapovanie funkcií. Všetky tieto klasifikácie môžu prebiehať vďaka použitiu dostatočného množstva trénovacích dát. Neurčitosti v trénovacích dátach, alebo chybné označenia malého množstva trénovacích dát nie sú vďaka vysokej tolerantnosti neurónových sietí žiadnym problémom. Avšak, problémom pre neurónové siete môže byť práca so symbolmi (Ao et al., 2010).

1.2.1 Hlboké učenie

Hlboké učenie si predstavujeme ako širšiu podoblasť strojového učenia, reprezentovanú hlbokými neurónovými sieťami, ktoré sa používajú na tri druhy učenia (Roopak et al., 2019):

- **Kontrolované učenie**

Cielom prvého je napodobnenie učení, aké prebieha aj pri ľuďoch. To si vieme predstaviť ako spracovávanie vstupných údajov, či signálov za účelom dosiahnutia cieľa. Vhodným príkladom je aj učenie dieťaťa rozoznávať jednotlivé objekty, napríklad rieku od jazera, či už na základe vzorov, dĺžok, farieb, polohy, či iných. V tomto prípade ide o tzv. kontrolované učenie (angl. *supervised*

learning), kedy odhadujeme neznáme na základe mapovania známych vstupov.

- **Nekontrolované učenie**

Druhým typom DL je tzv. algoritmus bez dozoru, často nazývané ako nekontrolované učenie (angl. *unsupervised algorithm*). Jeho príkladom je hádzanie šípok na terč. Namiesto naprogramovania striktného pohybu hádzania sa použije metóda tréningu, čím dlhšie trénuje a zdokonaľuje svoje hody na terč, tým bližšie je k požadovanému cieľu. Každým hodom sa totiž jeho kinematické stupne voľnosti nastavujú tak, aby šípka dosiahla bližší cieľ.

- **Semi-kontrolované učenie**

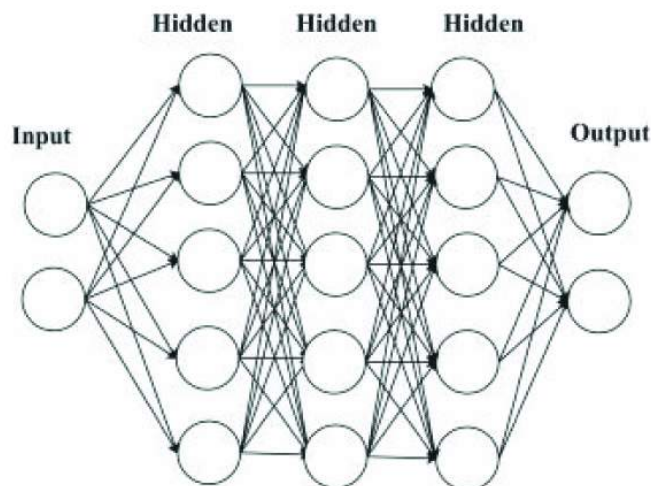
Tretím typom DL je tzv. semi-kontrolované učenie (angl. *semi-supervised learning*). Je používaný pri väčšine procesov v prírode a funguje na základe toho, že časť údajov je označená, no ostatné časti nie sú. Označená časť je pomocnou pri učení sa tej neoznačenej (El Naqa and Murphy, 2015).

Ako vidíme na Obrázku 1–2, neurónová sieť sa skladá z neurónov. Čiary, ktoré sa medzi nimi nachádzajú predstavujú prepojenia v podobe váh. Skladá sa z niekoľkých vrstiev, prvou je vstupná vrstva, znázornená na Obrázku 1–2 ako *Input*. Ďalej nasleduje určitý počet vrstiev nazývaných skryté vrstvy, vyobrazená ako *Hidden*. Pokiaľ je skrytých vrstiev väčší počet, vtedy je strojové učenie nazývané ako „hlboké“. Poslednou vrstvou je výstupná vrstva, známa tiež ako *Output* (Roopak et al., 2019).

Neurónové siete môžeme rozdeliť do troch skupín:

- **Dopredné neurónové siete**

Viacvrstvový perceptrón (angl. *Multilayer Perceptron*, skr. MLP) je taktiež nazývaný ako dopredná neurónová sieť (angl. *Feedforward Neural Network*, skr. FNN) a svojou štruktúrou je veľmi podobný jednovrstvovému perceptrónu (angl. *single-layer perceptron*, skr. SLP). Líši sa najmä tým, že obsahuje



Obrázok 1 – 2 Architektúra neurónovej siete. Zdroj: (Roopak et al., 2019)

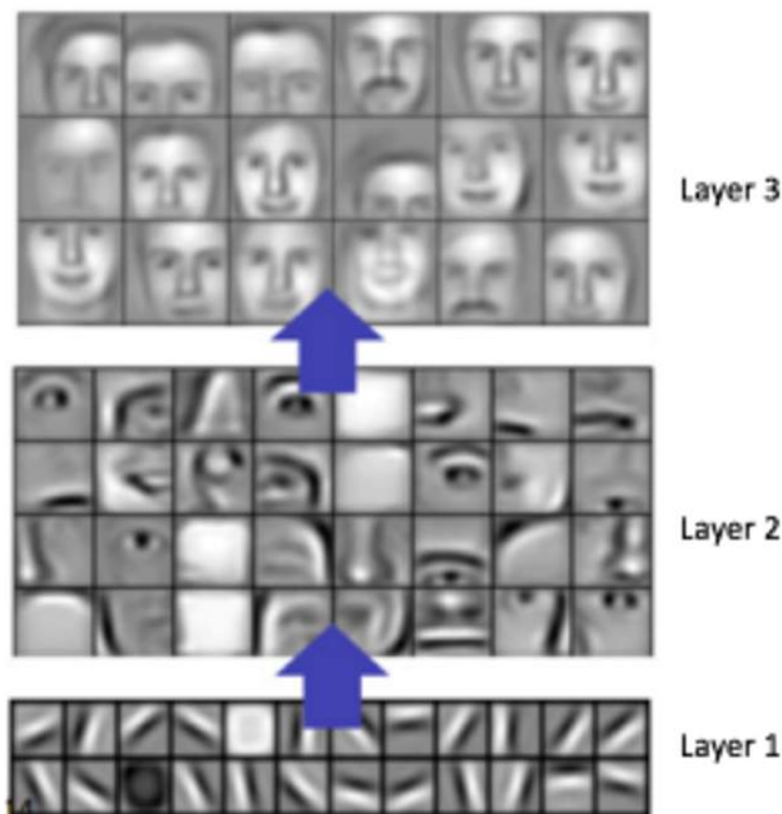
viacero vzájomne prepojených vrstiev, ktoré tvoria neurónovú sieť. Využíva aktivačné funkcie, ktoré môžu byť buď lineárne, alebo nelineárne. MLP a SLP sa líšia najmä na základe algoritmu spätného šírenia (angl. *back-propagation algorithm*). Ten dnes už predstavuje bežnú metódu tréningu neurónových sietí. Práve vďaka tomuto algoritmu dochádza k tomu, že sa chyby pozorované vo výstupnej vrstve prenášajú naspäť do vstupnej a na základe tejto informácie nastáva možnosť upraviť sieť MLP, ktorej architektúra je zobrazená aj na obrázku vyššie (Beysolow et al., 2017).

- **Konvolučné neurónové siete**

Konvolučná neurónová sieť (angl. *Convolutional Neural Network*, skr. CNN) je modelom hlbokého učenia a vznikla v roku 2012 (Sindagi and Patel, 2018). Práve tento model sa v dnešnej dobe používa najmä v oblastiach spracovania jazyka a obrazu a funguje na základe toho, že sa naučí priestorové vlastnosti, či už hrany, textúry a iné javy, ktoré vystihujú cieľovú skupinu.

CNN pracuje na základe vrstvenia. Klasifikácia obrázkov je rovnako, ako je znázornené na Obrázku 1 – 3, rozdelená do niekoľkých krokov. V prvej, spodnej

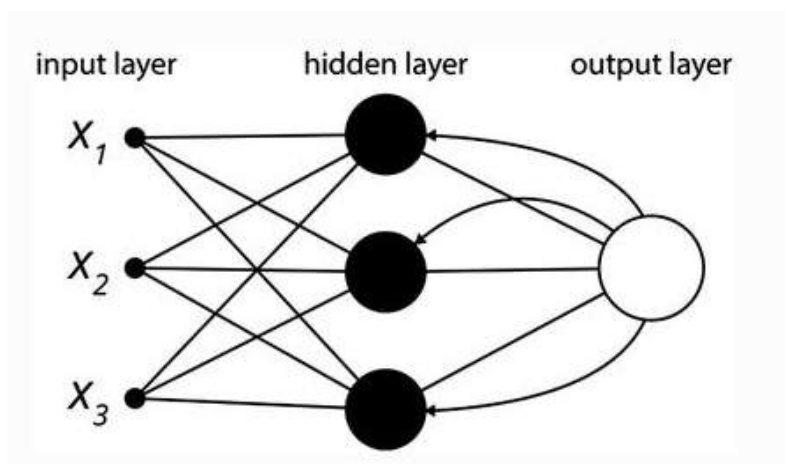
časti je len zachytenie samotných okrajov. V ďalšej, druhej časti sú tieto hrany spájané do jednoduchších objektov a tie sa napokon zhromažďujú v tretej časti do zložitejších celkov (Albawi et al., 2017).



Obrázok 1 – 3 Príklad klasifikácie obrázkov. Zdroj: (Albawi et al., 2017)

- **Rekurentné neurónové siete**

Rekurentné neurónové siete (angl. *recurrent neural networks*, skr. RNN) sú vyobrazením modelov umelej neurónovej siete, kde si dané vetvy a ich spojenia predstavujeme ako riadený cyklus. Za tým sa ukrývajú presne určené kroky po hranách množinou už použitých hrán. Takéto kroky sa špecificky používajú na spracovanie signálov, reči či textu. Znáznornenie tejto siete je aj bližšie popísané na Obrázku 1–4 (Beysolow et al., 2017).



Obrázok 1 – 4 Sieť RNN. Zdroj: (Beysolow et al., 2017)

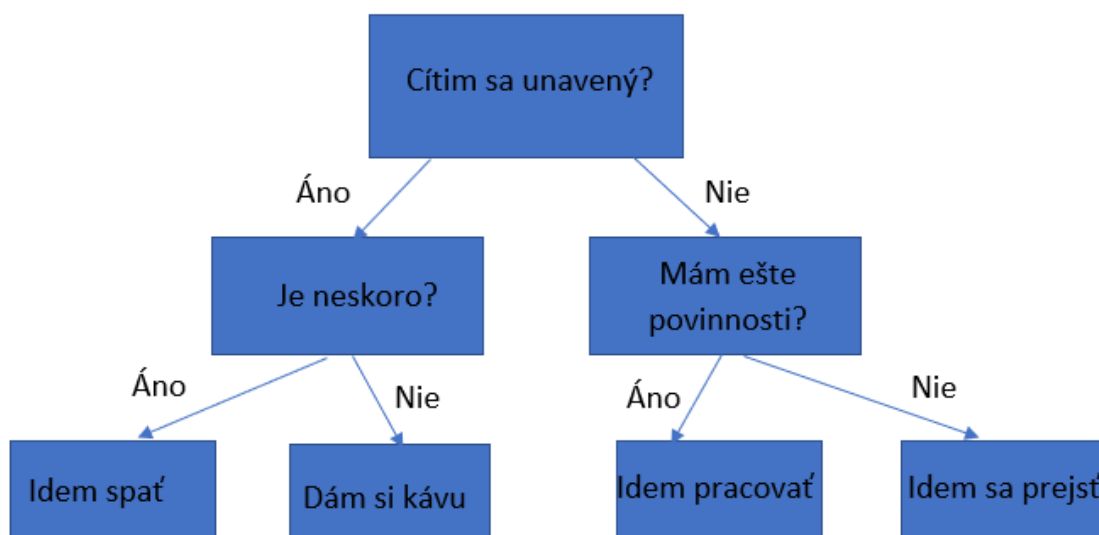
1.2.2 Štandardné metódy strojového učenia

Dnes už poznáme mnoho algoritmov, ktoré sa využívajú na strojové učenie. V danom oddieli sú bližšie predstavené niektoré z nich.

- **K-means algoritmus** je veľmi známou a často používanou metódou zhľukovania. Funguje na princípe vytvárania zhľukov takých prvkov, ktoré majú podobné vlastnosti. Prvky sú do jednotlivých tried zaraďované bez mien, čo odlišuje tento algoritmus od klasifikácie. K-means reprezentujú dva základné kroky. V prvom ide o priradenie centroidov, teda bodov definujúce zhľuky, do priestoru so zhľukmi. Počet centroidov sa ale musí rovnať počtu jednotlivých zhľukov. Druhým krokom je optimalizácia, ktorá posúva jednotlivé centroidy do prostriedku oblasti. Tam sa k nim opäť priradzujú ich najbližšie body. Po opakovaní týchto krokov sa centroidy odstránia a ostávajú už len finálne zhľuky, ktoré obsahujú podobné vlastnosti (Ripley, 2007).
- **Support vector machine** (skr. SVM) je ďalším algoritmom strojového učenia. Funguje na základe analýzy údajov pre klasifikáciu a taktiež regresnú analýzu. Tie ďalej triedi do dvoch skupín, teda určuje, do akej kategórie je vhodné zaradiť nový dátový bod (Steinwart and Christmann, 2008). SVM na-

šiel využitie najmä pri klasifikáciách obrázkov, textu či hypertextu, ale taktiež pri vedných odboroch alebo znakoch napísaných rukou.

- **Rozhodovacie stromy** (angl. *decision trees*) predstavujú binárny systém, ktorý hierarchicky a vo viacerých stupňoch rozhoduje, teda vyhodnocuje vyhovenie, resp. nevyhovenie určitej podmienke. Takýmito krokmi vyhodnocovania sa napokon dopracuje až ku triede, resp. riešeniu, ktoré je akceptovateľné. Tento postup je zobrazený aj na príklade Obrázku 1–5 a popisuje cestu od koreňa stromu k jeho listom. Tie obsahujú vyhodnotenie toho, čo bude nasledovať v blízkej budúcnosti na základe stavu, v akom sa vykonávateľ nachádza. Proces prechodu stavu sledujeme na tzv. uzloch. Kritériám v rozhodovacom strome sú priradované váhy, podľa ktorých sa určuje to, v akej postupnosti budú jednotlivé kritériá nasledovať. Je to tak najmä preto, že kritéria s najväčšou váhou dokážu dáta roztriediť do dvoch tried najlepšie (Michalski et al., 2013).



Obrázok 1 – 5 Príklad rozhodovacieho stromu. Zdroj: vlastné

- **Gradient boosting machine** (skr. GBM) je ďalšou technikou strojového učenia používanou na klasifikačné a regresné úlohy. Funguje na princípe zlep-

šovania tzv. slabých modelov prostredníctvom ich spájania do jedného silného prediktívneho modelu. GBM generuje predikčné modely, ktoré si môžeme predstaviť aj ako vyššie spomínané rozhodovacie stromy. Každý rozhodovací strom vie predpovedať chybu predchádzajúceho rozhodovacieho stromu a to zvyšuje, teda zlepšuje gradient (chybu) (Natekin and Knoll, 2013).

1.3 YOLO

V roku 2015 bol prvý krát prezentovaný algoritmus YOLO (angl. *You Only Look Once*, slov. pozri sa len raz), ktorý sa stal známy najmä na základe svojej presnosti a rýchlosti (Wang et al., 2018). Bol vyvinutý s cieľom zahrnutia jedнокrokového procesu detekcie a následnej klasifikácie. Tento algoritmus slúži na detekciu jednotlivých objektov na obrázkoch v reálnom čase a prvý krát ho vo svojom článku popísal Joseph Redmon s kolektívom (Redmon et al., 2016). Jeho využitie sa našlo už v mnohých smeroch, či už pri detekciách ľudí, zvierat, dopravy, či napríklad pri parkovacích automatoch. Algoritmus YOLO sa líši od ostatných najmä v tom, že sa predikcie ohraničení súčasne vytvárajú s predikciami tried (Thuan, 2021).

YOLO vzniklo ako najlepší predstaviteľ CNN, ktorý prišiel na základe naliehavej požiadavky na zlepšenie detekcie prostredníctvom jej zrýchlenia. Výkon CNN však nepostačoval na riešenie komplexnejších úloh, preto sa začalo pracovať na lepšom modeli, ktorý by zvládol riešiť úlohy efektívnejšie. YOLO nie len detekuje obrázky automaticky, ale taktiež poskytuje hodnotu pravdepodobnosti toho, ako si je istý detekovanými objektami (Yap et al., 2021).

Algoritmus rozdeľuje obrázky do mriežok, ktoré majú väčšinou rozmery 19×19 , jednotiek (Redmon et al., 2016). To zabezpečuje predikciu určitého počtu boxov od každej jej bunky, vďaka čomu je algoritmus predikcií tak rýchly. Následne je každej bunke priradená najpravdepodobnejšia trieda.

Pre lepšie nahliadnutie k výhodám a nevýhodám tohto algoritmu je potrebné jeho porovnanie s iným algoritmom. To je znázornené v nasledujúcej tabuľke na

Obrázku 1–6 ako porovnanie algoritmu YOLO a Fast R-CNN na databáze VOC 2007 (Wang et al., 2018). Fast R-CNN je algoritmus založený na jeho predchádzajúcej verzii, teda R-CNN, a vznikol ako jej zlepšenie (Girshick, 2015). To sa týka najmä rýchlosti tréovania a testovania dát, ale taktiež zahŕňa zlepšenie presnosti jednotlivých detekcií. Preto je vhodné porovnať YOLO algoritmus práve s týmto vylepšením konvolučných neurónových sietí.

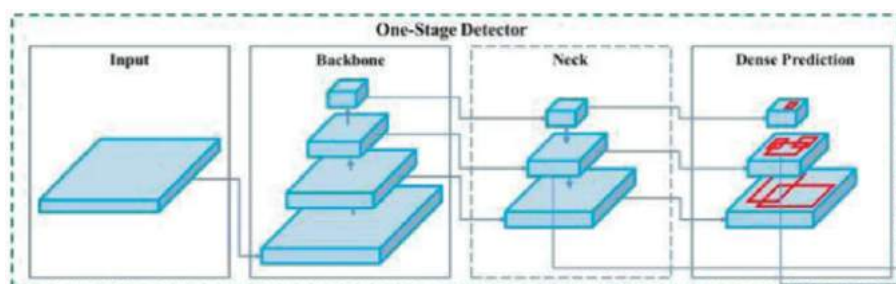
Type/Algorithm	YOLO (%)	Fast R-CNN (%)
Object Recognizes Precision IOU>0.5	67.50	69.40
Class Recognizes Precision 0.1<IOU<0.5	180	9.80
Class Recognizes Similarity IOU> 0.1	6.55	5.00
Class Fault IOU> 0.1	3.80	4.10
Background IOU<0.1	4.15	11.7

Obrázok 1–6 Porovnanie YOLO a Fast R-CNN. Zdroj: (Wang et al., 2018)

Z uvedeného obrázka vyplýva, že algoritmus YOLO dokáže podstatne lepšie rozlišovať kategórie. Napríklad, pri pravdepodobnosti, že R-CNN nezachytí žiadnu triedu na pozadí, vykazuje R-CNN až 11,7%, pričom YOLO predstavilo toto percento takmer trikrát nižšie (Wang et al., 2018).

YOLO už dnes obsahuje niekoľko rôznych verzií, medzi ktoré patria: (Redmon and Farhadi, 2017)

- YOLO v2: Základná verzia YOLO má dve podstatné chyby, a to nepresné určovanie polohy (angl. *inaccurate positioning*) a nízku mieru návratnosti (angl. *low recall rate*). Táto verzia sa preto venuje práve zlepšeniu týchto dvoch aspektov,



Obrázok 1 – 7 Architektúra YOLO V4. Zdroj: (Aly et al., 2021)

ktoré zabezpečujú väčšiu rýchlosť a kvalitu. Nie je to na základe prehĺbovania, či rozširovania siete, ale práve prostredníctvom jej zjednodušenia.

- YOLO V3: S príchodom každej novej verzie sa tvorcovia zameriavali na dosah presnejších výsledkov. Táto tretia verzia využíva na detekciu objektov siete pyramíd prvkov (angl. *Feature Pyramid Networks*, skr. FPN).
- YOLO V4: Narozdiel od verzie YOLO V3 prichádza opäť s presnejšími výsledkami detekcií. Taktiež zlepšuje chybovosť, ktorá je narozdiel od predošlej verzie nižšia. Pre detekciu objektov používa sieť PANet (angl. *Path Aggregation Network for Instance Segmentation*, slov. Sieť Agregácie Ciest pre Segmentáciu Inštancií), ktorá poskytuje rôzne úrovne detekcií a jeho architektúra je znázornená na Obrázku 1 – 7.
- YOLO V5: Táto doposiaľ najlepšia a najnovšia verzia prichádza s ešte lepším výkonom a presnosťou, preto sme sa v tejto práci rozhodli pokračovať s danou verziou YOLO. Jej bližší popis je znázornený v podkapitole nižšie.
- iné: Okrem týchto verzií existujú aj ďalšie, limitované verzie, ako napríklad YOLO-LITE.

1.3.1 YOLOv5

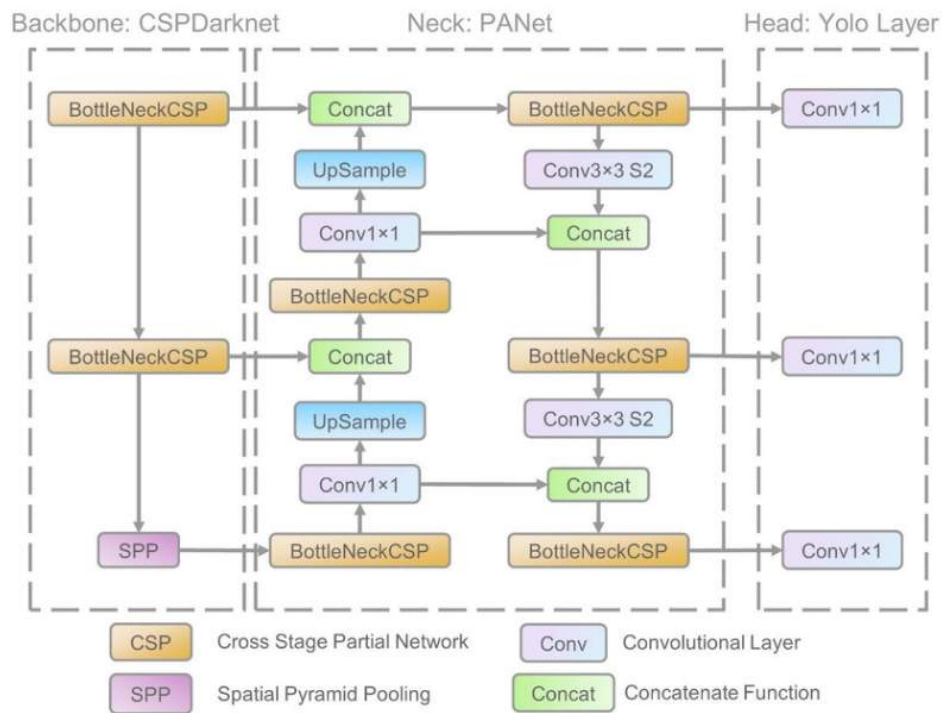
Ako bolo spomínané vyššie, YOLOv5 je jednou z generácií YOLO a prichádza s lepším výkonom, presnosťou a rýchlosťou a je jeho najnovšou verziou. Pôvodne vznikla

ako vylepšená verzia YOLO V3 pre PyTorch a jej prvú publikáciu uverejnila firma Ultralytics LLC v roku 2020 (Yap et al., 2021).

Architektúra YOLO V5 je zložená z troch základných častí, a to:

- **Backbone:** Vloženie dát do CSP Darknet (angl. *Cross Stage Partial*) kvôli extrakciám funkcií. Práve tu sa extrahujú vlastnosti objektov (Wang et al., 2020).
- **Neck:** V tejto časti dochádza k prevodu dát z CSP Darknet do PANet (angl. *Path Aggregation Network for Instance Segmentation*) (Liu et al., 2018). PANet zabezpečuje ukladanie vlastností detekcií do tzv. pyramíd, vďaka čomu sú identifikované objekty s inými vlastnosťami.
- **Head:** V poslednej časti sa prostredníctvom *Yolo Layer* vydávajú výsledky detekcií, ako napríklad trieda, lokácie, skóre, veľkosť či iné.

YOLO V5 architektúra je bližšie vyobrazená na Obrázku 1–8, ktorý popisuje jej vyššie uvedené časti.

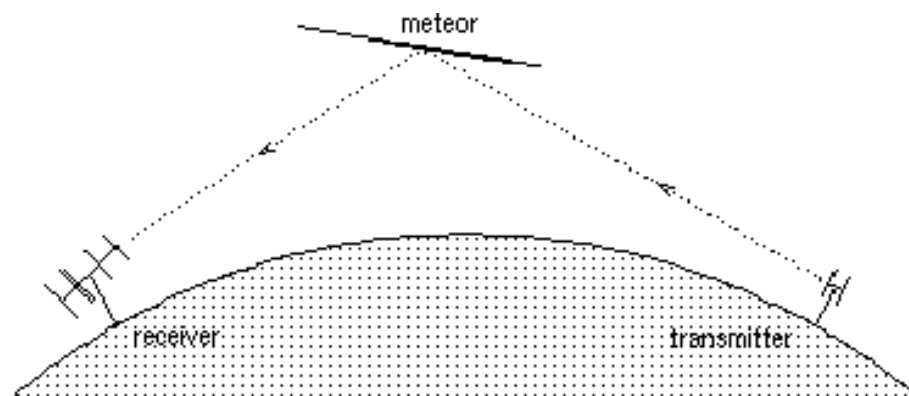


Obrázok 1 – 8 Architektúra YOLO V5. Zdroj: (Xu et al., 2021)

2 Detekcia meteorov na rádiových spektrogramoch

Pre danú prácu vytvorenia prostredia boli vybrané práve meteorické stopy, ktoré sú ďalej nazývané ako ozveny. Tieto stopy boli zachytené pomocou rádiového rozptylu, ktorý je efektívnou technikou na pozorovanie meteorov. Keď meteor vstúpi do atmosféry, jeho stopa môže odrážať rádiové vlny z vysielача do prijímača.

Pri zachytení meteoru je možné prijať malé časti vysielania z prostredia rádiových staníc vo vzdialenosti 500 – 2000km od miesta pozorovania. Avšak, priamy rádiový kontakt nie je možný kvôli zakriveniu Zeme (Wislez, 1996). Ako zber týchto dát funguje je možné vidieť na Obrázku 2–1. My sledujeme charakteristiku prijatého signálu, ktorá ďalej súvisí s fyzikálnymi parametrami meteorickej udalosti.

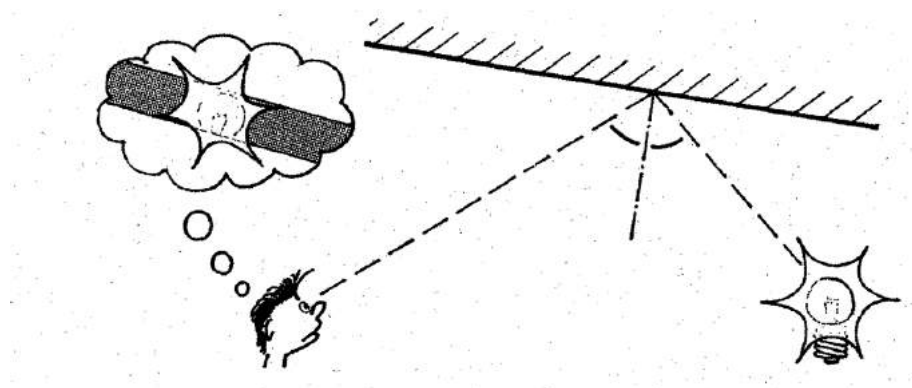


Obrázok 2–1 Signály z rádiového vysielача (angl. *transmitter*) sú do prijímača (angl. *receiver*) rozptýlené stopou meteoru, čo umožňuje detekciu meteoru. Zdroj: (Wislez, 1996)

Dôležitým faktom je, že zachytený odraz je zrkadlový. Preto si vysielач môžeme predstaviť ako žiarovku a meteor ako dlhé zrkadlo na Obrázku 2–2. To má za následok fakt, že meteory daného prúdu prinášajú odrazy len vo veľmi obmedzenom páse oblohy (Wislez, 1996).

2.1 Rozdelenie meteorov

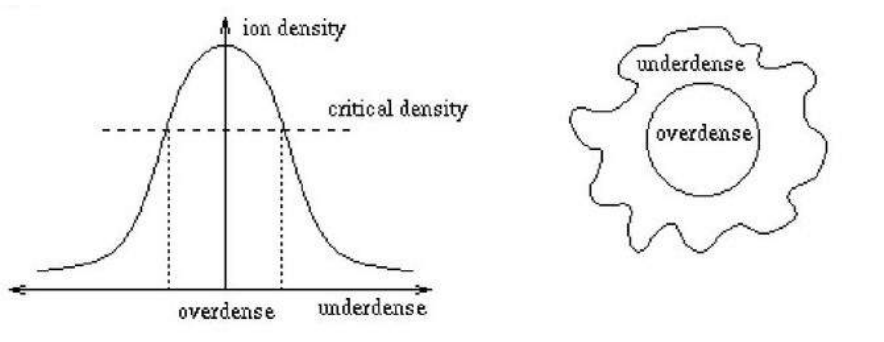
Odrážanie rádiových vln meteorom závisí od hustoty voľných elektrónov v jeho stope. Vo všeobecnosti sú známe dva základné prípady, a to veľmi nízke a veľmi vysoké elek-



Obrázok 2–2 Odraz rádiových vln na stopách meteorov je zrkadlový. Zdroj: (Wislez, 1996)

trónové hustoty meteoru. Meteory s nízkou hustotou budeme nazývať *Underdense* a naopak, tie s vysokou hustotou budú ďalej spomínané ako *Overdense*.

Plazmová frekvencia, ktorá je úmerná hustote iónov, vykazuje normálne rozdelenie. Stred stopy, kde je plazmová frekvencia vyššia ako použitá rádiová frekvencia, je *Overdense* plazma, viď Obrázok 2–3.



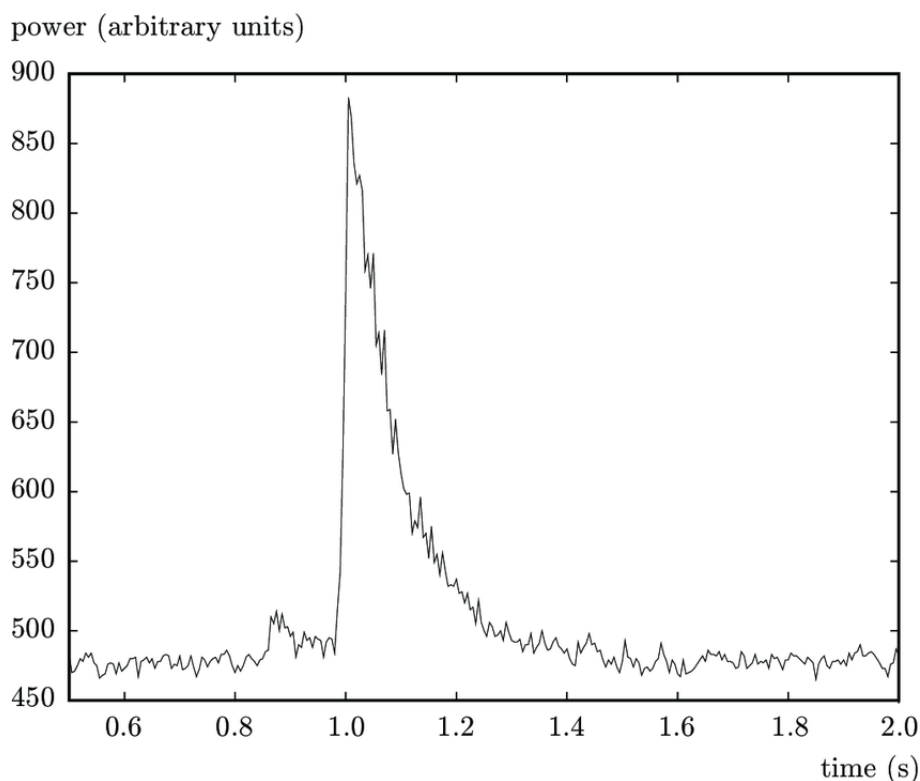
Obrázok 2–3 Rozloženie hustoty iónov v stope meteoru. Zdroj: (Wislez, 1996)

2.1.1 Nízkohustotné – *Underdense*

Nízkohustotné meteory predstavujú slabé, teda menej výrazné meteory. Ich hornou hranicou je približne 2×10^{14} elektrónov na meter. To značí hustotu čiar všeobecne spojenú s meteormi piatej magnitúdy.

Na Obrázku 2–4 môžeme vidieť zachytený signál prijatý odrazom na nedosta-

točne hustých stopách. Signály z týchto nedostatočne hustých meteorov v skutočnosti netrávajú dlhšie ako niekoľko desiatín sekundy. Tieto javy sú veľmi početné, čo umožňuje použitie štatistických redukčných techník.



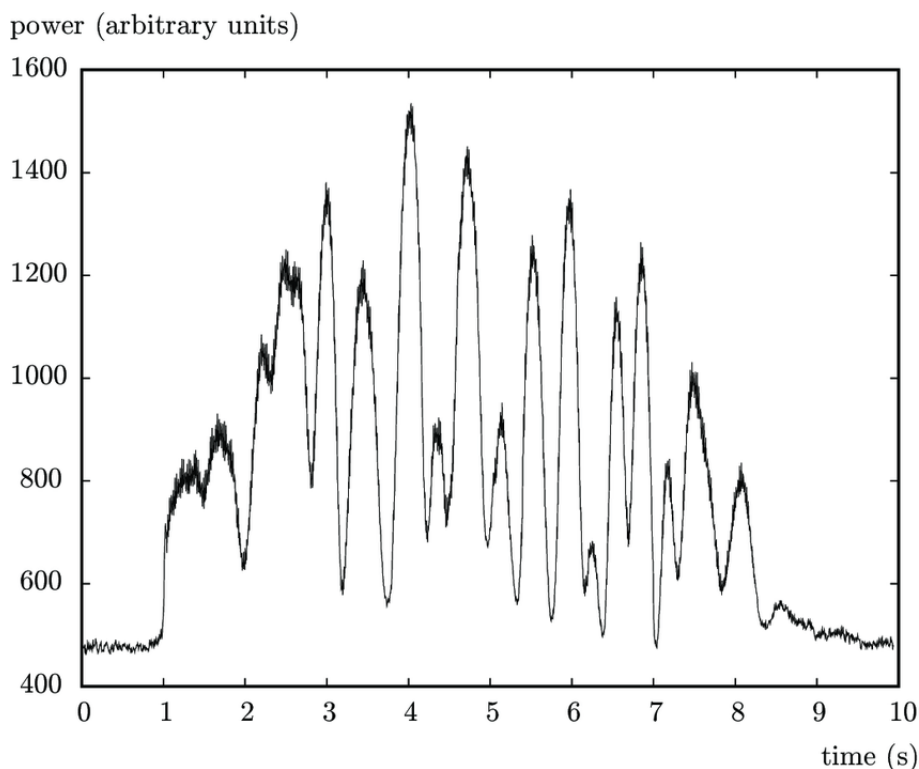
Obrázok 2–4 Typický meteor s nízkou hustotou, pozorovaný zariadením RAMSES. Zdroj: (Wislez, 1996)

2.1.2 Prehustené – *Overdense*

Tieto meteory majú nižšiu početnosť, ale predstavujú jasnejšie meteory. Ich odrazy môžu trvať až niekoľko sekúnd. To ale spôsobuje problémy, pretože počas dlhého časového obdobia môže dochádzať k deformáciám spôsobeným vetrom. To má za následok to, že príliš hustá stopa môže odrážať rádiové vlny do miesta prijímača, aj keď pôvodne nebola vhodne orientovaná na zrkadlový odraz. Spôsobuje to aj fakt, že vývoj výkonu signálu je väčšinou nepredvídateľný a často aj veľmi nepravidelný.

Na Obrázku 2–5 je ukázaná typická stopa pozorovania takéhoto typu meteoru.

Silné oscilácie na obrázku, čiže hlboké vyblednutie sú dôsledkom skreslenia stopy vetrom na rôznych častiach jeho trasy.

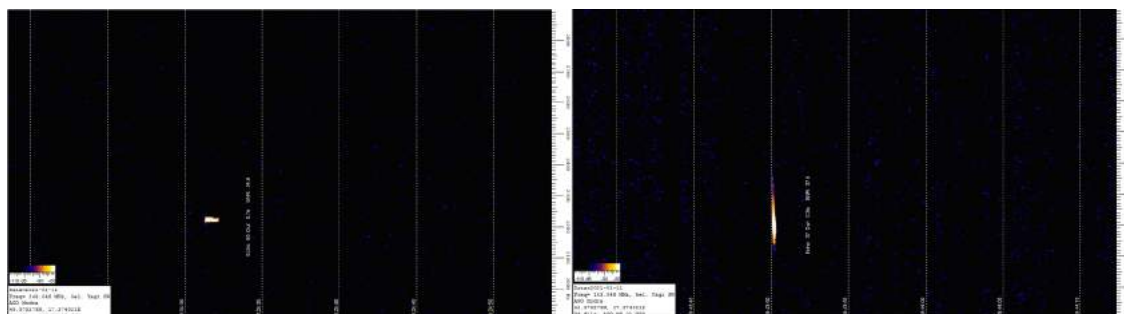


Obrázok 2–5 Typický meteor s prehustením, pozorovaný zariadením RAMSES. Zdroj: (Wislez, 1996)

2.1.3 Hlava ozveny – *Headecho*

Niektoré meteory, ktoré prejavujú veľmi jasnú stopu, môžu vykazovať *headecho*, teda „hlavu ozveny“¹. Dochádza k tomu vtedy, keď sa od plazmového oblaku, ktorý obklopuje meteorit vstupujúci do atmosféry, odrazia rádiové vlny. Vyskytujú sa najčastejšie vtedy, keď majú meteory veľmi nízky sklon dráhy k atmosfére. Plazmový oblak sa pohybuje spolu s meteoritom jeho rýchlosťou. Na Obrázku 2–6 je znázornené porovnanie ozveny bez headecha vľavo a ozveny s headechom v pravo, ktorý sa na dátach prejavuje zvislým natiahnutím po osi y (Wislez, 1996).

¹V ďalšom texte budeme používať anglický ekvivalent – *headecho*.



Obrázok 2–6 Porovnanie meteoru bez *head* s meteorom s *head*om. Zdroj: AGO Modra

Všetky získané prípady, s ktorými sa v tejto bakalárskej práci pracuje, boli zachytené prijímačom, ktorý sa nachádza na observatóriu **AGO Modra-Piesok**. Toto observatórium je profesionálnym výskumným a taktiež vzdelávacím pracoviskom, ktoré spadá pod Fakultu matematiky, fyziky a informatiky Univerzity Komenského v Bratislave (Zigo, 2018). V Európe taktiež existuje sieť rádio-meteorických pozorovaní, známa ako RMOB (angl. *Radio Meteor Observing Bulletin*), kde by sa mohlo využiť aj naše finálne riešenie. Viac o tejto sieti je možné dočítať sa na tomto linku².

2.2 Analýza súčasného stavu

Radar **GRAVES** (fran. *Grand Réseau Adapté à la Veille Spatiale*) je systém, ktorý bol vyvinutý vo Francúzsku a sledovaním sa zameriava na vesmírnu oblasť, konkrétne sa venuje detekcii meteorov a vesmírneho odpadu. Jeho vývoj trval takmer 15 rokov a prvý krát sa začal používať v septembri v roku 2005.

Funguje na základe **Dopplerovej** a **smerovej informácie** a využíva fázové pole štyroch vysielateľov. Na Obrázku 2–7 môžeme vidieť porovnanie sledovania meteoru zachytené na snímke z videa a jeho zobrazenie prostredníctvom radaru GRAVES (Fleet, 2015).

V súčasnosti sú dáta, s ktorými pracujeme od začiatku projektu vyhodnocované práve prostredníctvom tohto detekčného radaru. Už v úvode kontroly dát sme si všimli, že aktuálne používaný algoritmus na automatickú detekciu meteorov nepra-

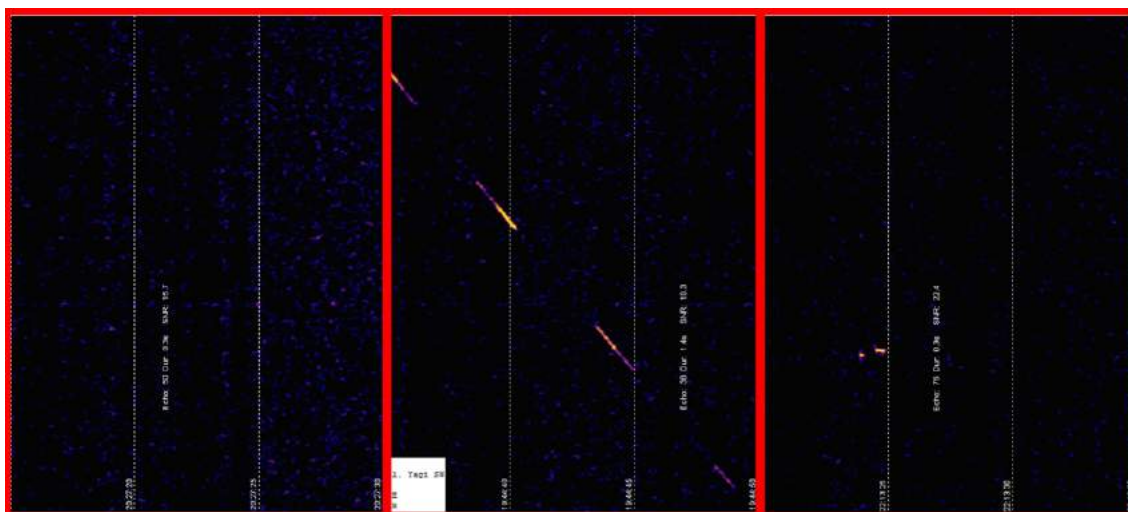
²<https://www.rmob.org/index.php>



Obrázok 2–7 Porovnanie snímky videa zachyteného meteoru a jeho zobrazenia prostredníctvom radaru GRAVES. Zdroj: (Fleet, 2015)

cuje správne, keďže mnohé označenia nezodpovedali hľadaným meteorom.

Na Obrázku 2–8 sú vyobrazené 3 prípady nesprávneho vyhodnocovania meteorov. V prvom prípade ide o označenie malej časti šumu, v druhom prípade o detekciu jednej časti družice a v treťom o chybné vyhodnotenie časovej dĺžky ozveny.



Obrázok 2–8 Príklady chybného detekovania meteorov GRAVES radarom. Zdroj: AGO Modra

Navyše, tento detekčný radar nedokáže rozoznať pri daných druhoch ozvien existenciu headecha, ani typ ozveny či už sa jedná o krátku alebo dlhú. Aj preto bolo

nevyhnutné navrhnúť riešenie, ktoré by správne rozoznávalo jednotlivé druhy ozvien, nedetekovalo falošné javy a priradzovalo im zodpovedajúcu dĺžku trvania.

3 Automatická detekcia meteorov na rádiových spektrogramoch pomocou metód hlbokého učenia

V rámci tejto časti bude popísaná praktická časť bakalárskej práce. Daná kapitola bude triedená do siedmych podkapitol, ktoré predstavujú jednotlivé fázy modelovania. Začínajúc pochopením úlohy a dát, kde bolo najdôležitejšie oboznámiť sa najmä s ich obsahom, cez ich následne spracovávanie a prípravu na modelovanie. V závere tejto kapitoly je bližšie popísané modelovanie, kde sme sa spätne vracali na kroky prípravy dát na modelovanie a celú kapitolu uzatvára predstavenie vyhodnotenia tejto práce, ktoré zahŕňa mnohé zaujímavé a veľmi užitočné poznatky.

3.1 Pochopenie úlohy

Na začiatku spolupráce s Fakultou matematiky, fyziky a informatiky Univerzity Komenského v Bratislave sme sa oboznámili s ich hlavným problémom spracovania meteorických ozvien a špecifikovali cieľovú úlohu. V ďalšom kroku sme získali dáta, s ktorými sme ďalej pracovali. Získali sme presne 3465 obrázkov, ktoré obsahovali rôzne druhy rádiových ozvien meteorického radaru, obsahujúcich pozitívne prípady (meteory), ako aj negatívne prípady (napríklad rôzne echá či odrazy z družíc, či iné falošné javy v kontexte úlohy). V rámci pozitívnych prípadov na nich boli zachytené meteory dvoch druhov, ktoré mohli, ale nemuseli obsahovať headecho, teda zvislé natiahnutie ozveny na osi y . Súčasťou taktiež boli obrázky neobsahujúce žiadny z uvedených javov.

Našou úlohou bolo vytvorenie modelu neurónovej siete, ktorý by dokázal automaticky detegovať hľadané javy na vstupných dátach. Tým by bolo možné uľahčiť čas a prácu ľuďom, ktorí tieto meteory doposiaľ hľadali ručne, prípadne používali menej úspešné automatické prístupy založené na jednoduchom prahovaní. Pre tvorbu nášho modelu sme si zvolili detekčnú neurónovú sieť YOLO.

Keďže dáta, ktoré sme v úvode práce obdržali neboli označené (anotované), bolo potrebné nájsť vhodné crowdsourcingové prostredie, ktoré po vytvorení projektu umožní dané dáta rýchlo a efektívne roztriediť do skupín a oddeliť od nepotrebných, chybných či zavádzajúcich dát.

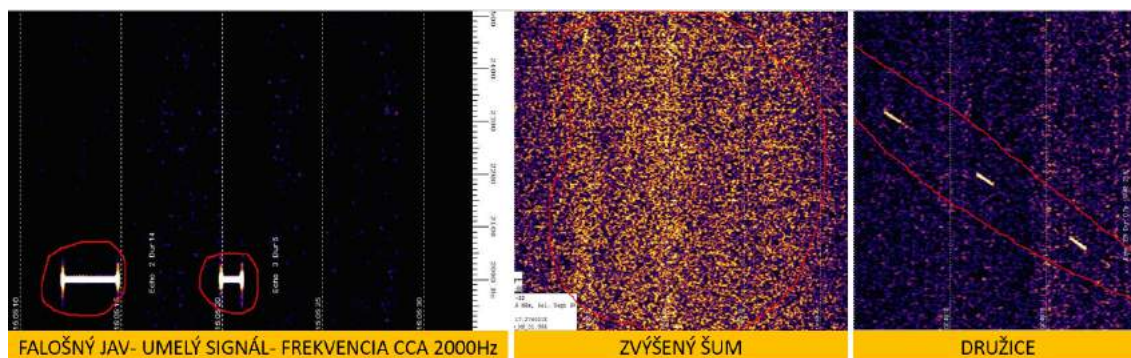
3.2 Pochopenie dát

Dáta nám neposkytujú len samotnú informáciu o existencii ozvien, ale aj o ich trvaní, dátume, frekvencii, parametroch, či o tom, na akom mieste bola daná snímka zhotovená. Pre nás sú však najdôležitejšie práve informácie ohľadom dĺžky trvania, ktorú si vieme zistiť analýzou pozície ozveny voči x -vej osi s pravidelným označením 5 sekundových intervalov v spodnej časti obrázka. Takisto je možné pomerne jednoducho určiť frekvencie ozvien, ktorá je viazaná na y -ovú os a jej hodnoty sa pohybujú od $1900Hz$ do $2900Hz$.

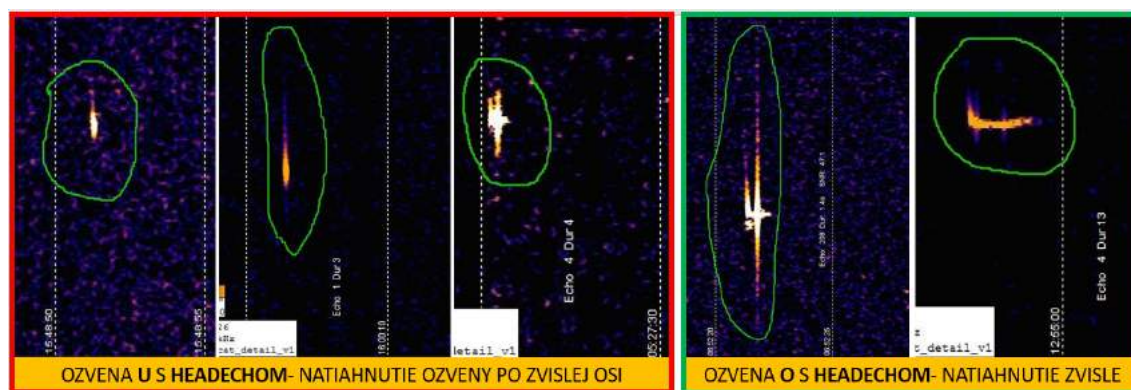
Dáta sme si v úvode rozdelili do štyroch skupín, ktoré sa líšili na základe dĺžky ozvien a existencii headecha. Všetky výskyty ozvien sa pozorovali pre frekvencie v rozmedzí $2200Hz - 2300Hz$.

Avšak, aj pri týchto dátach sme sa stretávali s rôznymi javmi, ktoré nepredstavovali hľadané meteory. Ako je znázornené na Obrázku 3–1, patrili medzi ne falošné javy, ako napríklad zachytený umelý signál. Práve pre tento druh falošného záznamu je typická alokácia ozvien mimo očakávaný rozsah frekvencií na osi y . Ozveny sa tak tiež v niektorých prípadoch nachádzali vo zvýšenom šume, ktorý nie len anotátorom, ale neskôr aj modelu sťažil ich detekciu. Na dátach boli zachytené aj rôzne druhy družíc. Pre tie bola typická prerušovaná línia a sklon z ľavej hornej strany do pravej dolnej.

Výskyt headecha na meteoroch je pozorovaný prostredníctvom ich zvislého natiahnutia v smere osi y . Na Obrázku 3–2 je zobrazené porovnanie výskytu rôznych druhov headecha na krátkych, nenasýtených ozvenách s dlhými, presýtenými ozvenami.



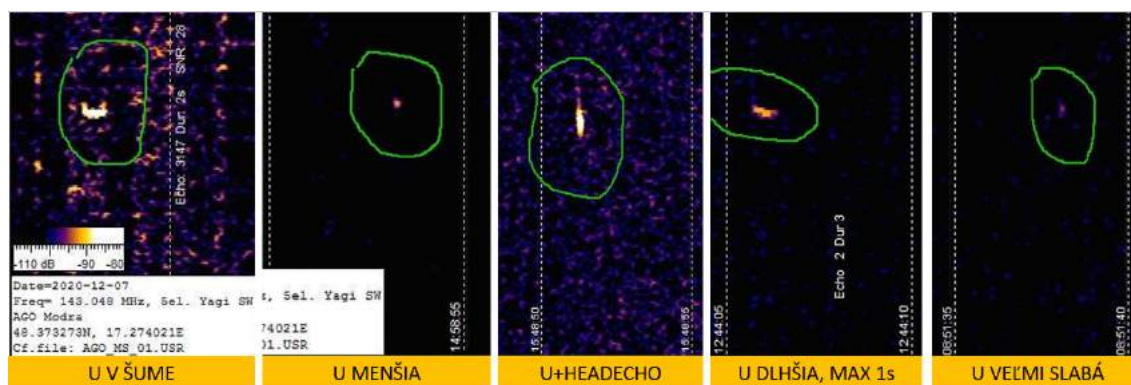
Obrázok 3 – 1 Príklad druhov vyskytujúcich sa falošných javov. Zdroj: P.Zigo, AGO Modra



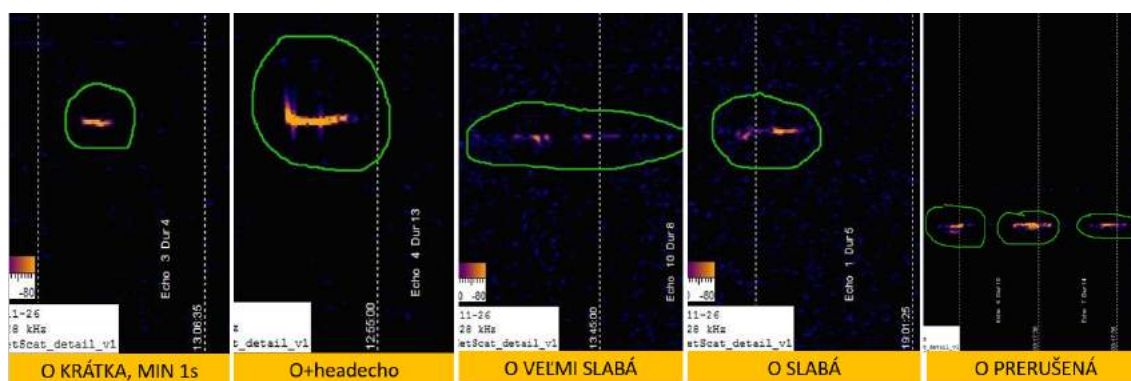
Obrázok 3 – 2 Príklad druhov headecha. Zdroj: P.Zigo, AGO Modra

Prvý druh meteorov sú krátke, tiež označované ako ozveny **typu U** (angl. *underdense*), ktoré sa môžu vyskytovať aj s obsahom headecha. Ich maximálna dĺžka trvania je do jednej sekundy. Pri krátkych a slabých ozvenách, ako je možné vidieť na Obrázku 3–3, môže dochádzať k nesprávnym označeniam či k ich zámene so šumom. Preto bolo potrebné venovať ich výskytu väčšiu pozornosť.

Druhou skupinou meteorov sú dlhé, nazývané tiež ozveny **typu O** (angl. *overdense*). Tie sa v mnohých prípadoch taktiež vyskytujú aj s headechom. Ako je znázornené na Obrázku 3–4, ich minimálna dĺžka je jedna sekunda.



Obrázok 3–3 Druhy krátkych ozvien U. Zdroj: P.Zigo, AGO Modra



Obrázok 3–4 Druhy dlhých ozvien O. Zdroj: P.Zigo, AGO Modra

3.3 Príprava dát v prostredí Zooniverse

Prostredie **Zooniverse** je vedecký portál vlastnený *Citizen Science Alliance* (Shuttleworth, 2016). Sídlí na Oxfordskej univerzite a Adlerovom planetáriu a má za sebou stovky projektov, vďaka čomu sa nie len dobrovoľníci môžu podieľať na crowdsourcingovom vedeckom výskume. To znamená, že ide o organizovanú činnosť, ktorá vedie k dosiahnutiu presne definovaných cieľov, a to s využitím väčšieho množstva zainteresovaných osôb.

Práve preto bolo toto prostredie najvhodnejším pre tento projekt. Zainteresovanými osobami, ktoré nám pomohli s triedením dát, boli študenti. Pripravením projektu v prostredí Zooniverse, v ktorom sú vysvetlené potrebné rozdiely medzi hľadanými javmi, ich následným vysvetlením na spoločnej online hodine a neskor-

ším anotovaním, boli dosiahnuté výsledky veľmi rýchlo a efektívne.

Práca v projekte Zooniverse je veľmi jednoduchá. Po zaregistrovaní sa je možné využiť linku *BUILD A PROJECT* (slov. vytvoriť projekt), ktorou otvoríme možnosť *CREATE A NEW PROJECT* (slov. vytvoriť nový projekt). Prostredie nás vzápätí žiada o vyplnenie projektového názvu, krátkeho popisu a úvodu.

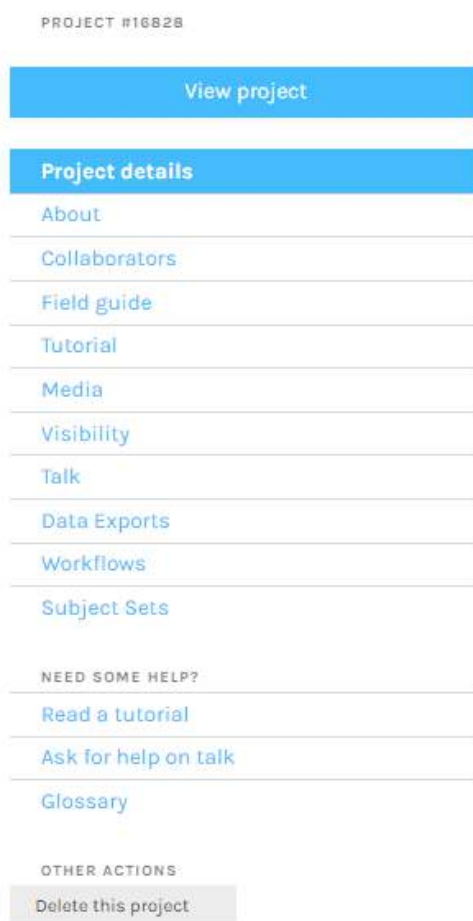
Úspešným uskutočnením daných podmienok sa dostaneme do prostredia, v ktorom sa budeme po zvyšok práce na projekte pohybovať. Všetky alternatívy nám umožňujú vykonať dôležité kroky pri vytváraní, či úprave projektu.

- **View:** Zobrazenie projektu je určená k otvoreniu už existujúceho projektu. Vďaka nej je možné skontrolovať, či daný projekt pracuje korektne už počas jeho výstavby.
- **Project details:** Presunutím sa nižšie na stránku detaily projektu získame možnosť úpravy zvoleného názvu, pridanie ďalších popisov, pridanie avatara projektu, či jeho pozadia.

Obrázok 3–5 znázorňuje pracovnú lištu, ktorá sa zobrazuje vľavo našej obrazovky. Hneď v úvode lišty je uvedené jedinečné ID nášho projektu. Kolónka *View project* (slov. zobrazenie projekt) je určená k otvoreniu už existujúceho projektu. Vďaka nej je možné skontrolovať, či daný projekt pracuje korektne už počas jeho výstavby.

Presunutím sa nižšie na stránku *Project details* (slov. detaily projektu) získame možnosť úpravy zvoleného názvu, pridanie ďalších popisov, pridanie avatara projektu, či jeho pozadia.

Ďalšia kolónka je veľmi dôležitá, keďže práve tá obsahuje základné informácie pre dobrovoľníkov. V jej popise boli uvedené základné rozdiely medzi hľadanými objektmi, chyby ktoré môžu počas anotovania nastať, rozmedzie a čas, v ktorom sa ozveny nachádzajú, alebo odkaz na krátky tutoriál, do ktorého sa môžeme kedykoľvek počas práce pozrieť. Ten je možné vytvoriť v kolónke *Tutorial* (slov. tutoriál) pridaním názvu a krokov s popismi.



Obrázok 3–5 Lišta prostredia Zooniverse pri vytváraní a úprave projektu

Ďalšou dôležitou časťou je *Visibility* (slov. viditeľnosť), v ktorej je nutné prepnúť nastavenie viditeľnosti projektu zo súkromnej na verejnú a z rozvíjajúcej sa na aktívnu.

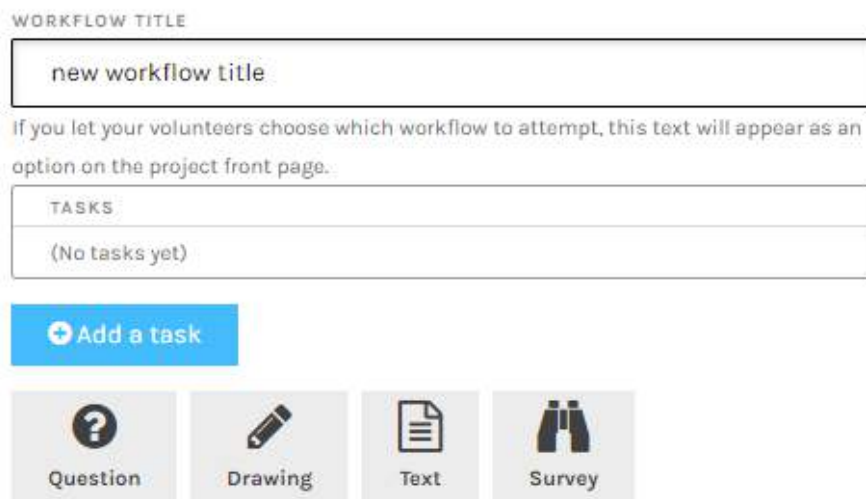
Všetky dáta, ktoré budú anotované, je potrebné nahráť do okna *Subject Sets* (slov. súbory predmetov). Po ich spracovaní je možné raz denne požiadať o export upravených dát v okne *Data Exports* (slov. export dát).

3.3.1 Pracovné prostredie projektu

Avšak, najdôležitejším oknom je *Workflow* (slov. pracovné prostredie). Všetky potrebné zmeny aj samotné vytvorenie projektu sa uskutočňuje práve tu. Po jeho

rozkliknutí a zvolení *New workflow* (slov. nové pracovné prostredie) si vytvárame úlohy prostredníctvom možnosti *Add a task* (slov. pridaj úlohu).

Ako môžeme vidieť na Obrázku 3–6, vybrať si vieme zo štyroch možností úloh. Pre vytvorenie úloh projektu anotovania ozvien boli najmä využité otázky z dôvodu pretriedenia dát a úlohy zamerané na kreslenie, pre získanie orezanej ozveny a jej parametrov, tiež nazývaných *bounding box* (slov. ohraničujúci box). Každú úlohu je možné ďalej nastaviť podľa potreby.



Obrázok 3–6 Možnosti voľby úlohy v prostredí Zooniverse. Zdroj: vlastné

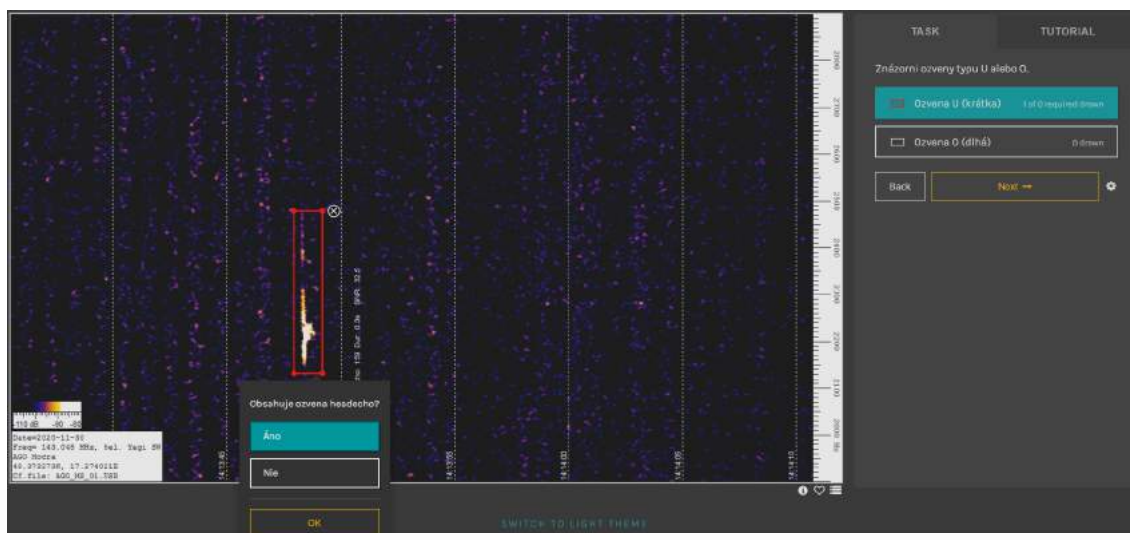
Toto okno nám umožňuje vykonať aj dôležité nastavenia týkajúce sa napríklad počtu opakovaní jednej snímky, to znamená, koľko krát sa môže jedna snímka zobraziť dobrovoľníkom. Taktiež umožňuje pridať do existujúceho pracovného prostredia vytvorené tutoriály, alebo sprístupniť nahraté dáta. Po jeho úspešnom vytvorení sa celé pracovné prostredie otestuje prostredníctvom tlačítka *Test this workflow* (slov. otestuj tento pracovný tok).

3.3.2 Zooniverse z pohľadu anotátora

Po úspešnom splnení všetkých potrebných krokov na vytvorenie anotačného projektu sa anotátori prostredníctvom linku³, ktorý je stále dostupný, mohli dostať priamo do prostredia anotovania. Po jeho začatí prostredníctvom tlačidla *Get started* (slov. začať) sa anotátorom postupne zobrazujú jednotlivé dáta, ako je znázornené na Obrázku 3–7.

V prvom kroku odpovedajú na otázku, či zobrazený obrázok obsahuje ozvenu. V prípade, že odpovedajú kladne, postupujú ďalej, kedy si po zvolení typu nachádzajúcej sa ozveny vytvoria okolo nej rámček. Ten sa vyskytuje pri ozvenách krátkych ako červený a pri dlhých ako biely. Hneď po označení ozveny sa anotátorovi zobrazí potreba zvoliť jej vlastnosť, a to či obsahuje, alebo neobsahuje headecho.

Ak anotátor označil všetky javy vyskytujúce sa na danom obrázku, zvolí krok *Next* (slov. ďalej). Program sa ho kvôli možným nejasnostiam opýta či si je svojim krokom naozaj istý a chce pokračovať. V prípade nejasností mu ponúkne možnosť prekliknutia sa na tutoriál vpravo hore, alebo sa vrátiť o krok späť cez tlačidlo *Back* (slov. späť). Po uložení svojich krokov sa mu automaticky vygeneruje ďalší snímok.



Obrázok 3–7 Prostredie Zooniverse viditeľné pre anotátorov. Zdroj: vlastné

³<https://www.zooniverse.org/projects/kamilajencikova/ozveny>

3.4 Extrakcia anotovaných dát z prostredia Zooniverse

Na spracovanie dát v prostredí Zooniverse pracovalo 33 študentov, ktorí dohromady vykonali 5758 anotácií. Tieto anotácie sa uložili ako výstup anotačného projektu do súboru .csv, ktorý je určený na ukladanie dát do tabuliek.

Takto pripravenú tabuľku bolo potrebné očistiť od nepotrebných dát, spracovať získané dáta, skontrolovať a následne odstrániť chybné údaje.

Na tieto kroky spracovania dát bol vybraný programovací jazyk Python a prostredie JupyterLab. V prvom kroku bolo potrebné daný .csv súbor načítať v programe. Časť jeho prvotného vyobrazenia je znázornená na Obrázku 3–8.

	classification_id	user_name	user_id	user_id_workflow_id	workflow_name	workflow_version	created_at/dt	metadata	annotations	subject_data	subject_ids
721	378795066	kamielijankova	2355346	2396972c19e7e690e759ce	20291	ozveny	28.31 2021-11-26 20:49:28 UTC	{\"translation_id\":\"51358\";\"tutorial\":{\"value\":\"\"};\"value\":\"\"};131_085013_233.jpg}	70511557		
732	378795076	-23967d9e7e690e759ce	23967d9e7e690e759ce	23967d9e7e690e759ce	20291	ozveny	28.31 2021-11-26 20:49:28 UTC	{\"translation_id\":\"51358\";\"tutorial\":{\"value\":\"\"};\"value\":\"\"};0201_013816_38.jpg}	70511631		
733	378795087	kamielijankova	2355346	2848e72c19e7e68ea0f6	20291	ozveny	28.31 2021-11-26 20:49:31 UTC	{\"translation_id\":\"51358\";\"tutorial\":{\"value\":\"\"};\"value\":\"\"};227_083958_439.jpg}	70511861		
734	378795092	-5f6e84f342d4397a8dec	5f6e84f342d4397a8dec	5f6e84f342d4397a8dec	20291	ozveny	28.31 2021-11-26 20:49:35 UTC	{\"translation_id\":\"51358\";\"tutorial\":{\"value\":\"\"};\"value\":\"\"};0227_091229_15.jpg}	70511771		
735	378795113	-23967d9e7e690e759ce	23967d9e7e690e759ce	23967d9e7e690e759ce	20291	ozveny	28.31 2021-11-26 20:49:44 UTC	{\"translation_id\":\"51358\";\"tutorial\":{\"value\":\"\"};\"value\":\"\"};131_085013_233.jpg}	70511557		
736	378795121	-23967d9e7e690e759ce	23967d9e7e690e759ce	23967d9e7e690e759ce	20291	ozveny	28.31 2021-11-26 20:49:48 UTC	{\"translation_id\":\"51358\";\"tutorial\":{\"value\":\"\"};\"value\":\"\"};227_083958_439.jpg}	70511861		
737	378795124	kamielijankova	2355346	23967d9e7e690e759ce	20291	ozveny	28.31 2021-11-26 20:49:50 UTC	{\"translation_id\":\"51358\";\"tutorial\":{\"value\":\"\"};\"value\":\"\"};201_090059_273.jpg}	70511673		
738	378795125	-23967d9e7e690e759ce	23967d9e7e690e759ce	23967d9e7e690e759ce	20291	ozveny	28.31 2021-11-26 20:49:52 UTC	{\"translation_id\":\"51358\";\"tutorial\":{\"value\":\"\"};\"value\":\"\"};220_181817_379.jpg}	70511713		
739	378795130	-5f6e84f342d4397a8dec	5f6e84f342d4397a8dec	5f6e84f342d4397a8dec	20291	ozveny	28.31 2021-11-26 20:49:54 UTC	{\"translation_id\":\"51358\";\"tutorial\":{\"value\":\"\"};\"value\":\"\"};228_190224_307.jpg}	70511699		
740	378795145	kamielijankova	2355346	2348e72c19e7e68ea0f6	20291	ozveny	28.31 2021-11-26 20:50:00 UTC	{\"translation_id\":\"51358\";\"tutorial\":{\"value\":\"\"};\"value\":\"\"};10111_204716_54.jpg}	70511289		
741	378795153	-23967d9e7e690e759ce	23967d9e7e690e759ce	23967d9e7e690e759ce	20291	ozveny	28.31 2021-11-26 20:50:03 UTC	{\"translation_id\":\"51358\";\"tutorial\":{\"value\":\"\"};\"value\":\"\"};31330_113828_99.jpg}	70511203		
742	378795158	-5f6e84f342d4397a8dec	5f6e84f342d4397a8dec	5f6e84f342d4397a8dec	20291	ozveny	28.31 2021-11-26 20:50:06 UTC	{\"translation_id\":\"51358\";\"tutorial\":{\"value\":\"\"};\"value\":\"\"};31130_113828_99.jpg}	70511203		
743	378795166	kamielijankova	2355346	2348e72c19e7e68ea0f6	20291	ozveny	28.31 2021-11-26 20:50:11 UTC	{\"translation_id\":\"51358\";\"tutorial\":{\"value\":\"\"};\"value\":\"\"};131_110228_291.jpg}	70511570		
744	378795176	-23967d9e7e690e759ce	23967d9e7e690e759ce	23967d9e7e690e759ce	20291	ozveny	28.31 2021-11-26 20:50:17 UTC	{\"translation_id\":\"51358\";\"tutorial\":{\"value\":\"\"};\"value\":\"\"};10111_204716_54.jpg}	70511289		
745	378795186	-23967d9e7e690e759ce	23967d9e7e690e759ce	23967d9e7e690e759ce	20291	ozveny	28.31 2021-11-26 20:50:28 UTC	{\"translation_id\":\"51358\";\"tutorial\":{\"value\":\"\"};\"value\":\"\"};227_013206_108.jpg}	70511792		
746	378795198	-5f6e84f342d4397a8dec	5f6e84f342d4397a8dec	5f6e84f342d4397a8dec	20291	ozveny	28.31 2021-11-26 20:50:29 UTC	{\"translation_id\":\"51358\";\"tutorial\":{\"value\":\"\"};\"value\":\"\"};131_085013_233.jpg}	70511557		
747	378795204	kamielijankova	2355346	2848e72c19e7e68ea0f6	20291	ozveny	28.31 2021-11-26 20:50:32 UTC	{\"translation_id\":\"51358\";\"tutorial\":{\"value\":\"\"};\"value\":\"\"};227_083958_432.jpg}	70511858		
748	378795206	-23967d9e7e690e759ce	23967d9e7e690e759ce	23967d9e7e690e759ce	20291	ozveny	28.31 2021-11-26 20:50:35 UTC	{\"translation_id\":\"51358\";\"tutorial\":{\"value\":\"\"};\"value\":\"\"};131_124730_326.jpg}	70511580		
749	378795212	-5f6e84f342d4397a8dec	5f6e84f342d4397a8dec	5f6e84f342d4397a8dec	20291	ozveny	28.31 2021-11-26 20:50:38 UTC	{\"translation_id\":\"51358\";\"tutorial\":{\"value\":\"\"};\"value\":\"\"};227_110936_560.jpg}	70511881		
750	378795225	kamielijankova	2355346	23967d9e7e690e759ce	20291	ozveny	28.31 2021-11-26 20:50:47 UTC	{\"translation_id\":\"51358\";\"tutorial\":{\"value\":\"\"};\"value\":\"\"};0227_091229_15.jpg}	70511771		
751	378795237	-23967d9e7e690e759ce	23967d9e7e690e759ce	23967d9e7e690e759ce	20291	ozveny	28.31 2021-11-26 20:50:56 UTC	{\"translation_id\":\"51358\";\"tutorial\":{\"value\":\"\"};\"value\":\"\"};227_083314_432.jpg}	70511858		
752	378795250	-5f6e84f342d4397a8dec	5f6e84f342d4397a8dec	5f6e84f342d4397a8dec	20291	ozveny	28.31 2021-11-26 20:51:02 UTC	{\"translation_id\":\"51358\";\"tutorial\":{\"value\":\"\"};\"value\":\"\"};201_090059_273.jpg}	70511673		
753	378795260	kamielijankova	2355346	2848e72c19e7e68ea0f6	20291	ozveny	28.31 2021-11-26 20:51:07 UTC	{\"translation_id\":\"51358\";\"tutorial\":{\"value\":\"\"};\"value\":\"\"};130_213841_262.jpg}	70511236		
754	378795261	-23967d9e7e690e759ce	23967d9e7e690e759ce	23967d9e7e690e759ce	20291	ozveny	28.31 2021-11-26 20:51:07 UTC	{\"translation_id\":\"51358\";\"tutorial\":{\"value\":\"\"};\"value\":\"\"};131_110228_291.jpg}	70511570		
755	378795268	-5f6e84f342d4397a8dec	5f6e84f342d4397a8dec	5f6e84f342d4397a8dec	20291	ozveny	28.31 2021-11-26 20:51:11 UTC	{\"translation_id\":\"51358\";\"tutorial\":{\"value\":\"\"};\"value\":\"\"};227_083958_439.jpg}	70511861		
756	378795271	kamielijankova	2355346	2396972c19e7e690e759ce	20291	ozveny	28.31 2021-11-26 20:51:16 UTC	{\"translation_id\":\"51358\";\"tutorial\":{\"value\":\"\"};\"value\":\"\"};227_084710_450.jpg}	70511867		
757	378795280	kamielijankova	2355346	2848e72c19e7e68ea0f6	20291	ozveny	28.31 2021-11-26 20:51:21 UTC	{\"translation_id\":\"51358\";\"tutorial\":{\"value\":\"\"};\"value\":\"\"};227_100004_497.jpg}	70511873		
758	378795283	-5f6e84f342d4397a8dec	5f6e84f342d4397a8dec	5f6e84f342d4397a8dec	20291	ozveny	28.31 2021-11-26 20:51:23 UTC	{\"translation_id\":\"51358\";\"tutorial\":{\"value\":\"\"};\"value\":\"\"};227_051939_298.jpg}	70511824		
759	378795292	kamielijankova	2355346	2348e72c19e7e68ea0f6	20291	ozveny	28.31 2021-11-26 20:51:31 UTC	{\"translation_id\":\"51358\";\"tutorial\":{\"value\":\"\"};\"value\":\"\"};227_062047_316.jpg}	70511835		
760	378795310	-23967d9e7e690e759ce	23967d9e7e690e759ce	23967d9e7e690e759ce	20291	ozveny	28.31 2021-11-26 20:51:46 UTC	{\"translation_id\":\"51358\";\"tutorial\":{\"value\":\"\"};\"value\":\"\"};227_110936_560.jpg}	70511881		

Obrázok 3–8 Príklad časti .csv výstupu z prostredia Zooniverse. Zdroj: vlastné

Popis jednotlivých stĺpcov:

- **classification_id**: Jedinečná ID každej vykonanej anotácie.
- **user_name**: Zobrazenie mena prihláseného používateľa, ktorý anotáciu vykonával. Ak išlo o neprihláseného používateľa, je tam uvedený reťazec identifikujúci jeho reláciu (angl. session).

-
- **user_id**: Každý zaregistrovaný používateľ má svoju jedinečnú ID, inak je tento údaj prázdny.
 - **user_ip**: IP adresa počítača dobrovoľníka.
 - **workflow_id**: Jedinečné ID projektu.
 - **workflow_name**: Názov projektu.
 - **workflow_version**: Verzia projektu.
 - **created_at**: Vyobrazenie dátumu a času vykonania anotácie.
 - **metadata**: Zobrazenie šírky a výšky obrázka a podrobných informácií o tom, kedy dobrovoľník začal a ukončil anotovanie konkrétneho obrázka, z akého počítača boli dané anotácie vykonané alebo v akom prehliadači daný používateľ pracoval.
 - **annotations**: Táto časť dát je tá najdôležitejšia pre získanie vstupných dát na tréning modelu. Skladá sa z troch základných skupín. Prvou je výstup na otázku „Vidíš na obrázku ozvenu?“. V prípade kladnej odpovede sú druhou časťou odpovede na otázku „Znázorni ozvenu typu U alebo O“, ktoré obsahujú súradnice x a y , poradové číslo anotácie, zaznamenanú šírku a dĺžku znázornenia, typ ozveny, ktorý môže byť buď Ozvena O (dlhá), alebo Ozvena U (krátka) a detail, kde hodnote 1 pripadá ozvena bez headecha a hodnote 0 ozvena s headechom. Tretou časťou je odpoveď na otázku „Si si istý svojou odpoveďou? Ak nie, použi tutoriál.“, ktorej odpoveď môže byť vo výstupe iba kladná.
 - **subject_data**: V tejto časti je najdôležitejšou informáciou názov obrázka, na ktorom bola daná anotácia vykonaná. Je to pôvodný názov dát získaných na vstupe.
 - **subject_ids**: ID predmetu.

3.5 Príprava dát na modelovanie

Celá príprava dát na modelovanie prebiehala v programovacom jazyku Python. V prvom kroku sa zo získaného výstupu v podobe .csv súboru odstránili nepotrebné stĺpce (`workflow_name`, `workflow_id`, `workflow_version`, `user_ip`, `user_id`, `user_name`, `classification_id`, `created_at`, `gold_standard`, `expert`). V ďalšom sme ošetrili dáta od chybných záznamov. Ošetrovanie bolo podmienené celkovou dĺžkou riadku, v ktorom anotátor zaznačil, že existuje ozvena, ale žiadnu graficky neoznačil. V takom prípade sa dĺžka záznamu výrazne odlišovala od toho plnohodnotného. Na základe tejto informácie bol takýto záznam z dát odstránený, nakoľko bez konkrétneho grafického vyznačenia ozveny nebol vhodný pre ďalšie použitie.

Dodatočne sa odstránili aj záznamy ozvien, ktoré anotátori síce označili, ale ich označenie nesedelo vzhľadom ich vlastnostiam. Takéto chybné anotácie boli nájdené **ručným prehľadávaním** dát a malo zamedziť neskoršiemu chybnému učeniu neurónovej siete. Vytvorenie kódu pre vykreslenie anotácií, ktoré sa uložili v podobe obrázkov do priečinku, uľahčilo ručné prehľadávanie dát vďaka ich vizualizácii.

Obrázok 3–9 znázorňuje časť upravenej výstupnej tabuľky, ktorá bola vytvorená na základe dát z .csv súboru. Táto tabuľka obsahuje 8 stĺpcov, a to:

- **Event**: Stĺpec s názvom obrázkov.
- **Subject_id**: Jedinečné ID pre každý obrázok.
- **X**: X-ová súradnica nájdenej anotácie, ktorá predstavuje stred anotovanej ozveny na osi x.
- **Y**: Y-ová súradnica nájdenej anotácie, ktorá predstavuje stred anotovanej ozveny na osi y.
- **Šírka**: Šírka rámčeka okolo anotovanej ozveny.
- **Výška**: Výška rámčeka okolo anotovanej ozveny.
- **Headecho**: Stĺpec obsahujúci informácie ohľadom výskytu headecha.

- `Typ_ozveny`: Typ ozveny, ktorý predstavuje jeden zo štyroch možných druhov.

	event	subject_id	x	y	sirka	vyska	headecho	typ_ozveny
189	event20210227_064750_328.jpg	69136039	421.3125	411	29	26	nie	Ozvena U (krátka)
190	event20210226_225029_749.jpg	69135617	425.2650146484375	412.0185546875	32.532073974609375	27.326934814453125	nie	Ozvena U (krátka)
191	event20210201_060349_184.jpg	69134649	415.3125	426	32	29	ano	Ozvena U (krátka)
192	event20201130_095849_53.jpg	69134897	1090.138671875	451.9330749511719	15.67822265625	53.06454467734375	nie	Ozvena O (dlhá)
193	event20210112_234339_874.jpg	68902833	1083.89892578125	424.5231018066406	72.865966796875	34.848907470703125	ano	Ozvena O (dlhá)
194	event20210201_001858_9.jpg	69134474	415.4363708496094	313.8355285644531	51.74456787109375	159.45773315429688	ano	Ozvena U (krátka)
195	event20210131_054257_137.jpg	69134109	426.83349609375	413.8192443847656	36.180023193359375	30.1500244140625	nie	Ozvena U (krátka)
196	event20210131_104015_280.jpg	69134252	415.45867919921875	413.7779541015625	37.6162109375	27.90875244140625	ano	Ozvena U (krátka)
197	event20210112_134645_375.jpg	68901947	419.3125	447	22	19	nie	Ozvena U (krátka)
198	event20210112_201522_516.jpg	68902183	430.3125	437	24	34	nie	Ozvena U (krátka)
199	event20210131_074159_207.jpg	69134179	845.25	415.9666748046875	36	34	nie	Ozvena U (krátka)
200	event20210227_012448_101.jpg	69135812	440.7806701660156	419.2366638183594	17.9521484375	14.7841796875	nie	Ozvena U (krátka)
201	event20210201_001027_5.jpg	69134470	428.98040771484375	367.39959716796875	30.889373779296875	100.39044189453125	ano	Ozvena U (krátka)
202	event20210201_104318_302.jpg	69134749	421.2013854980469	417.4913024902344	60.96697988046875	45.06255103515625	nie	Ozvena O (dlhá)
203	event20210226_235652_838.jpg	69135706	425.2650146484375	413.3198547363281	22.121795654296875	20.82049560546875	nie	Ozvena U (krátka)
204	event20210226_235652_838.jpg	69135706	668.6048583984375	414.6211242675781	28.628173828125	27.326934814453125	nie	Ozvena U (krátka)
205	event20210227_004310_45.jpg	69135756	419.3274841308594	415.66424560546875	30.889373779296875	21.2364501953125	nie	Ozvena U (krátka)
206	event20210131_115654_309.jpg	69134281	430.3125	393	22	63	ano	Ozvena U (krátka)
207	event20210226_234346_821.jpg	69135689	420.3125	410	26	30	nie	Ozvena U (krátka)
208	event20210227_032757_195.jpg	69135906	871.6334228515625	544.902099609375	38.0164794921875	15.84014892578125	nie	Ozvena O (dlhá)
209	event20210112_143955_398.jpg	68901982	419.59576416015625	439.149169921875	24.1202392578125	26.532257080078125	nie	Ozvena U (krátka)

Obrázok 3 – 9 Príklad časti vytvorenej výstupnej tabuľky. Zdroj: vlastné

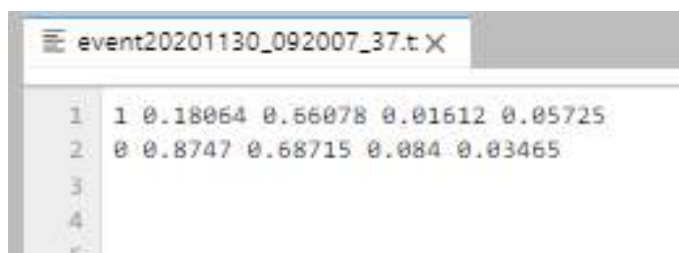
Z celkového počtu dát bolo týmto spôsobom zachytených a ošetrených 3870 pozitívnych anotácií. V Tabuľke 3 – 1 je zobrazené rozdelenie jednotlivých počtov podľa ich typu.

Anotácie (pozitívne)	Krátke	Dlhé
S headechom	1352	185
Bez headecha	2160	173

Tabuľka 3 – 1 Tabuľka s rozdeleným počtom jednotlivých druhov ošetrených anotácií

V ďalšej časti spracovania dát pre vytvorenie modelu YOLO bol vytvorený kód, ktorý spracúva potrebné informácie do textových súborov. Ten na základe vytvorenej výstupnej tabuľky vytvorí nový riadok nového textového súboru. Každému anotovanému obrázku s pozitívne zachytenými javmi prislúcha práve jeden textový súbor. V prípade viacerých ozvien zachytených na jednom obrázku sa informácie o ďalšej ozvene uložia do nového riadku textového súboru.

Ako môžeme vidieť na Obrázku 3–10, každý takýto riadok obsahuje 5 informácií numerického charakteru o type a lokácii nájdenej ozveny. Prvé číslo predstavuje druh ozveny. Číslo 0 reprezentuje krátku ozvenu bez headecha, číslo 1 naopak krátku ozvenu s headechom. Pod číslom 2 neurónová sieť získa informáciu o nachádzajúcej sa dlhej ozvene bez headecha a v poslednej skupine pod číslom 3 o dlhých ozvenách s headechom.



Obrázok 3–10 Príklad textového súboru pripraveného z výstupných dát. Zdroj: vlastné

Ďalšie dve číselné informácie zobrazujú x -ovú a y -ovú súradnicu. Tieto súradnice predstavujú krajný bod ohraničenej ozveny a boli upravené podľa potrieb na číselnú škálu 0–1. Pre súradnicu x sa vypočítajú pomocou vzorca 3.1, kde sa sčíta stredová súradnica *bounding boxu* na osi x s polovičnou hodnotou jeho šírky a následne sa predelí šírkou daného obrázka (angl. *picture width*). Podobným spôsobom je pre súradnicu y použitý vzorec 3.2, ktorý spočítava stredovú súradnicu *bounding boxu* na osi y s polovičnou hodnotou jeho výšky, a to celé sa následne predelí výškou daného obrázka (angl. *picture height*).

$$X = \frac{x + \frac{width}{2}}{picture_width} \quad (3.1)$$

$$Y = \frac{y + \frac{height}{2}}{picture_height} \quad (3.2)$$

Poslednými dvomi je šírka a výška, ktoré predstavujú vzdialenosť od stredu obrázka po jeho okraj a sú taktiež prepočítané na škálu 0–1.

Keďže sa pri anotovaní objavili niektoré dáta anotátorom viac krát, uložili sa aj viac krát medzi pozitívne zachytené javy. Preto bolo potrebné vytvorenie kódu, ktorý tieto dáta očistil od duplicitne sa vyskytujúcich. Následne z celkového počtu pozitívne zachytených javov sa počet jedinečných znížil na 2023 obrázkov.

Tie bolo potrebné z dôvodu nasledujúceho strojového učenia rozdeliť na tri základné skupiny, predstavujúce **trénováciu** : **validačnú** : **testovaciu** časť.

Prvou skupinou bola **trénovacia množina**, ktorá slúži na prvotné tréovanie, alebo ináč nazývané učenie modelu neurónovej siete. Obsahuje najväčšiu časť dát, ktorá v konkrétnom prípade zaberá takmer 72%. To predstavuje presne 1453 vstupných obrázkov.

Druhou skupinou je **validačná množina**, ktorá slúži na overenie toho, ako sa daný program dokázal naučiť potrebné informácie. Predstavuje časť dát, ktorú model vidí počas učenia a následne sa na nej otestuje. V tomto prípade predstavovala približne 8% všetkých dát, čo bolo presne 162 obrázkov.

Poslednou skupinou je **testovacia množina**, ktorá predstavuje časť dát viditeľnú pre model prvý krát až pri testovaní. Na základe týchto výsledkov vieme porovnať, či je model dostatočne naučený a pripravený na nasadenie. Je druhou najväčšou skupinou a predstavuje 20% dát, čo je 408 obrázkov.

3.6 Modelovanie

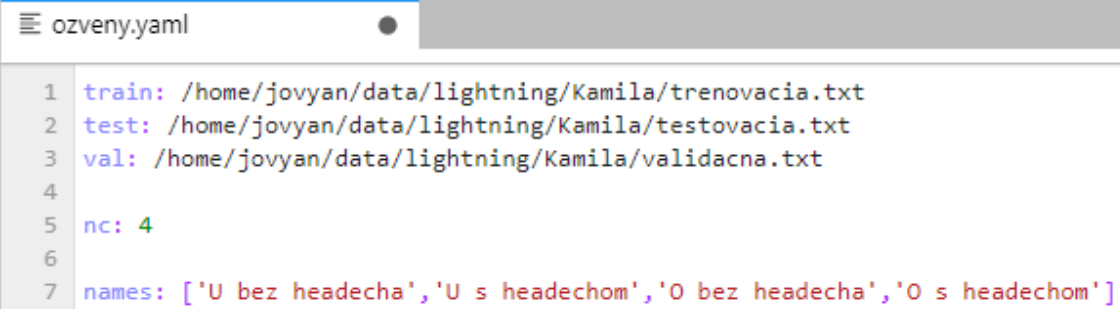
Po spracovaní dát a ich úspešnej príprave na modelovanie prostredníctvom detekčnej neurónovej siete YOLO sa v úvode použil programovací jazyk Python, konkrétne terminál JupyterLab.

Na začiatku učenia modelu bolo potrebné prostredníctvom trénovacej množiny neurónovú sieť natréovať. Boli vyskúšané mnohé verzie tréovania s cieľom nájsť také, ktoré dokáže neurónovú sieť naučiť detegovať jednotlivé prípady s čo najnižšou chybovosťou. Pre tréovanie najlepšieho modelu sa napokon použila kombinácia parametrov, ktoré sú znázornené aj na Obrázku 3–11.

```
!python yolov5/train.py --img 609 --rect --batch 64 --epochs 400 --data /home/jovyan/data/lightning/Kamila/ozveny.yaml  
--cfg /home/jovyan/data/lightning/Kamila/yolov5/models/yolov5s.yaml --save-period 40 --workers 0
```

Obrázok 3–11 Príklad kódu tréovania neurónovej siete. Zdroj: vlastné

Samotnému tréovaniu dát predchádzalo vytvorenie yaml súboru potrebného pre uloženie jednotlivých ciest k detailom o tréovacích, validačných a testovacích dátach. Ako je znázornené na Obrázku 3–12, yaml súbor obsahoval informácie aj o počte tried, ktorý bol zapísaný prostredníctvom `nc: 4` a taktiež o ich názvoch, znázornených ako `names: ['U bez headecha', 'U s headechom', 'O bez headecha', 'O s headechom']`.



```
ozveny.yaml  
1 train: /home/jovyan/data/lightning/Kamila/trenovacia.txt  
2 test: /home/jovyan/data/lightning/Kamila/testovacia.txt  
3 val: /home/jovyan/data/lightning/Kamila/validacna.txt  
4  
5 nc: 4  
6  
7 names: ['U bez headecha', 'U s headechom', 'O bez headecha', 'O s headechom']
```

Obrázok 3–12 Zobrazenie yaml súboru s cestami k množinám. Zdroj: vlastné

Popis parametrov volania funkcie pre tréovanie YOLO:

- `!python`: Týmto príkazom sa nastavuje spôsob spúšťania nášho súboru.
- `yolov5/train.py`: Táto časť predstavuje odkaz na súbor s pripravenými dátami na tréning.
- `--img 609`: V danom prípade testovania je zadaná hodnota polovica veľkosti dát. Nižšia veľkosť bola nastavovaná z dôvodu nepostačujúceho výkonu hardvéru.
- `--rect`: Aby sme zabezpečili obdĺžnikové ohraničenia z pohľadu modelu, je dôležité použiť tento príkaz. V inom prípade by sa podľa zadanej súradnice model pozeral na dané prípady ako na štvorec.

- `--batch 64`: Časť *batch size* určuje počet, kombinácie koľkých obrázkov prechádzajú cez neurónovú sieť v tom istom čase. V tomto prípade sme zvolili číslo 64.
- `--epochs 400`: Ďalšia časť *epoch* predstavuje to, koľko krát prejdú neurónovou sieťou dáta z trénovacej množiny.
- `--data`: V tejto časti je známa cesta k **yaml** súboru, ktorý obsahuje vyššie spomínané informácie.
- `--cfg`: Inou časťou trénovacieho kódu je cesta k **yaml** súboru, ktorý obsahuje informácie o modeli.
- `--save-period 40`: V tejto časti sa nastavuje perióda, ktorá ukladá podľa zvolenej hodnoty, v našom prípade je to hodnota 40 to, po koľkých epochách sa jednotlivé váhy budú ukladať.
- `--workers 0`: Nastavenie počtu pracujúcich na tomto projekte. Z dôvodu umožnenia zadania väčšieho čísla v časti `--batch` sa hodnota **workers** musela nastaviť na nulu.

Výsledné trénovanie sa zastavilo na počte 198 zo 400 *epoch*. Zastavené bolo z toho dôvodu, že za posledných 100 *epoch* model nezaznamenal žiadne zlepšenie pri učení. Daný počet *epoch* sa trénoval vyše 2,5 hodiny. Trénovanie prebiehalo na **GPU Tesla K40c**, čo predstavuje radu výpočtových grafických procesorov spoločnosti NVIDIA. Vďaka CUDA technológii sme mohli využiť vysoký výkon grafickej karty nie len pre urýchlenie testovania. Výsledky sa uložili do šiestich rôznych priečinkov, medzi ktoré patrili trénovania **best** (slov. najlepšie), **epoch40**, **epoch80**, **epoch120**, **epoch160** a trénovanie **last** (slov. posledné). Všetky spomínané trénovania bolo potrebné otestovať v ďalšom kroku tvorby modelu.

V testovacej časti (pre získanie detekcií na vyhodnotenie) sme použili kód, ktorý je znázornený na Obrázku 3–13. Pri testovaní dát bolo nutné vyskúšať rôzne kombinácie nastavení kódu z dôvodu nájdenia optimálneho riešenia vhodným nastavením

citlivosti detekcie. Tieto kombinácie sa týkali použitia rôznych natrénovaných množín a hodnôt parametrov IoU a `conf`. Avšak, najlepšie možné riešenie bolo nájdené práve na dátach `best`, s parametrom *Intersection over Union* (skr. IoU) nastaveným na 0,3 a `conf` 0,24. Vtedy sa ukazovalo výsledné porovnanie množstva skutočne pozitívnych, falošne pozitívnych, skutočne negatívnych a falošne negatívnych ako najvhodnejšie.

```
!python yolov5/detect.py --source /home/jovyan/data/lightning/Kamila/testovacia
--weights yolov5/runs/train/exp16/weights/best.pt --iou 0.3 --conf 0.24 --img 609 --save-txt --save-conf
```

Obrázok 3 – 13 Zobrazenie kódu pre testovanie dát. Zdroj: vlastné

Popis jednotlivých častí kódu testovania modelu:

- `!python`: Týmto príkazom sa nastavuje spôsob spúšťania nášho súboru.
- `--source`: Táto časť predstavuje odkaz na súbor s pripravenými dátami na testovanie.
- `--weights`: V tejto časti sa nachádza cesta k prípadom `best`.
- `--iou 0.3`: Parameter IoU (angl. *Intersection over Union*) predstavuje jednu z dvoch najdôležitejších častí, ktoré je potrebné nastaviť tak, aby najlepšie vyhovovali modelu. Tento parameter sa zameriaval na porovnávanie jednotlivých boxov tried daných predikcií. Vypočítal sa ako podiel prieniku boxov a ich zjednotenia.
- `--conf 0.24`: Druhým najdôležitejším parametrom je *confidence threshold*. V tomto prípade sme si nastavili minimálnu akceptovateľnú hodnotu pravdepodobnosti zaradenia do triedy. Znamená to, že sa všetky nájdené hodnoty nižšie ako 0,24, teda 24% nezarátajú.
- `--img 609`: Rovnako ako pri tréningovej časti, aj v testovacej sa nastavuje veľkosť obrázka, ktorú sme zvolili ako jeho polovičnú veľkosť, z dôvodu nedostatočného výkonu hardvéru.

- `--save-txt`: Veľmi dôležitá časť, ktorá výsledky testovania neukladá v podobe obrázkov, ale aj v podobe textového súboru. Ten obsahuje informácie o type ozveny a jeho parametroch.
- `--save-conf`: Vďaka tomuto príkazu sa do textového súboru uloží ešte jedna informácia, ktorá špecifikuje desatinné vyjadrenie percenta, akým si je model istý, že sa jedná o danú ozvenu zvoleného typu.

3.7 Vyhodnotenie riešenia

Aby bolo nájdené optimálne riešenie, bolo potrebné skúšať rôzne kombinácie parametrov `IoU` a `conf`, a taktiež rôzne trénovacie množiny. Výrazne sa od všetkých ostatných množín odlišovala práve tá, ktorú aj model vyhodnotil ako najlepšie natreňovanú a uložil ju pod názvom `best`. Rôzne parametre pri testovaní na týchto prípadoch ukázali, že pri použití príliš vysokého percenta parametra `conf` nepostačoval počet nájdených ozvien. Teda, z celkového počtu 408 prípadov zachytil ozvenu pri nastavení `conf` na 0,30 len na 384.

Pri finálnych nastaveniach dokázal model rozoznať až 392 prípadov z celkového počtu 408. Avšak, na mnohých z nich sa vyskytovala ozvena viac ako jedenkrát, preto je bližšie porovnanie reálneho vstupu a predikcií modelu znázornené v Tabulke 3–2, kde:

- TP (angl. *true positive*) sú skutočne pozitívne hodnoty, čo predstavuje počet, koľko pozitívnych javov skutočne predikoval daný model.
- FN (angl. *false negative*) sú falošne negatívne hodnoty, ktoré mali byť predikované ako pozitívne, ale model ich nezachytil.
- FP (angl. *false positive*) sú falošne pozitívne hodnoty, ktoré model označil ako pozitívne, ale pritom sa nejednalo o ozveny.
- TN (angl. *true negative*) sú skutočne negatívne hodnoty, čo znamená, že model správne nezachytil javy, ktoré ani zachytiť nemal. Toto číslo bolo vyhodno-

cované na základe obrázkov, ktoré neobsahovali žiadne javy, a teda ktoré aj daný model nepredikoval pre žiadne druhy ozvien. Reálne však model dokázal rozpoznať očakávané udalosti a správne nepredikovať omnoho viac javov, či už sa jednalo o šum, družice, alebo iné.

Tabuľka 3–2 Kontigenčná tabuľka klasifikácie

Reálne hodnoty				
Predikcie		Pozitívne	Negatívne	
	Pozitívne	TP = 496	FP = 12	
Negatívne	FN = 16	TN = 10	26	
	512	22		

Hodnoty tabuľky za čiarou predstavujú súčty jednotlivých tried. Celkový počet pozitívnych javov bol 512, negatívnych 22. Naopak, celkový súčet predikcií pozitívnych javov bol 508 a negatívnych 26.

Vyhodnotenie tréningovej časti sa na jeho konci uložilo do niekoľkých skupín, ktoré je možné vidieť znázornené v Tabuľke 3–3. Počty *Labels* (slov. označenia) pre jednotlivé triedy predstavujú sumár toho, koľko krát sa v testovaných prípadoch našli ozveny určitého typu. Stĺpec *P* znázorňuje presnosť (angl. *precision*). Ide o percentuálne vyjadrenie pomeru predikovaných hodnôt, ktoré boli vyhodnotené správne (angl. *true positive*, skr. TP) a všetkých, ktoré sa v danej množine nachádzajú. Vypočítajú sa podľa 3.3.

$$P = \frac{TP}{TP + FP} \quad (3.3)$$

Ďalší stĺpec označený ako *R* predstavuje metriku návratnosti (angl. *recall*). Tá znázorňuje percento podielu skutočne pozitívnych a všetkých vykonaných anotácií, ako je zobrazené v 3.4.

Tabuľka 3–3 Sumarizácia validácie modelu

Class	Images	Labels	P	R	mAP	F1-score
All	162	187	0,608	0,768	0,666	0,679
U without headecho	162	114	0,635	0,728	0,648	0,678
U with headecho	162	59	0,566	0,78	0,755	0,656
O without headecho	162	6	0,494	0,815	0,501	0,615
O with headecho	162	8	0,726	0,75	0,761	0,738

$$R = \frac{TP}{TP + FN} \quad (3.4)$$

Stĺpec mAP určuje strednú priemernú presnosť (angl. *mean average precision*). Táto metrika porovnáva skutočné hodnoty *bounding boxov* (slov. ohraničujúci box) s tými detekovanými a tak vypočítava skóre. Čím je hodnota mAP vyššia, tým je model presnejší v daných detekciách. Predstavuje takzvaný kompromis medzi hodnotami P a R a taktiež zohľadňuje falošne pozitívne aj falošne negatívne detekcie. Vypočíta sa podľa 3.5, ktorý v prvom kroku zistí priemernú presnosť (angl. *average precision*, skr. AP) pre každú jednotlivú i -tú triedu a potom sa vypočíta ich priemer zo všetkých N sledovaných tried.

$$mAP = \frac{1}{N} \sum_{i=1}^N AP_i \quad (3.5)$$

Poslednou metrikou v Tabuľke 3–3 je metrika F1-skóre (angl. *F1-score*). Tá zobrazuje model z jeho celkovej správnosti. Vypočíta sa podľa 3.6 a predstavuje harmonický priemer medzi metrikami P a R.

$$F1 \text{ skóre} = 2 * \frac{P * R}{P + R} \quad (3.6)$$

Celkovým výstupom danej práce je okrem modelu zhrnutie jednotlivých anotácií vo forme výstupnej tabuľke po aplikácii procesu modelovania a vyhodnotenia. Tá

je znázornená na Obrázku 3–14. Daná tabuľka obsahuje 12 stĺpcov s dôležitými informáciami o nájdenej ozvene. Patria medzi ne:

- **event_name**: Znázornenie názvu obrázka, na ktorom sa ozvena nachádza.
- **label**: Typ obrázka z pohľadu jeho dĺžky a výskytu headecha. V tejto skupine sa môžu nachádzať vyššie spomínané štyri druhy ozvien.
- **x**: Súradnica X predstavuje stred *bounding boxu* ozveny na osi *x*.
- **y**: Súradnica Y predstavuje stred *bounding boxu* ozveny na osi *y*.
- **width**: Tento údaj znázorňuje dĺžku od súradnice X až po okraj *bounding boxu* nájdenej ozveny v pixeloch na súradnici *x*.
- **height**: Naopak, údaj *height* (slov. výška) je znázornením dĺžky od stredu ozveny až po okraj *bounding boxu* na súradnici *y*.
- **confidence**: V tejto časti je zachytené presvedčenie modelu. Dané hodnoty sú ako desatinné vyjadrenie percenta.
- **length_in_sec**: Dĺžka ozveny v sekundách bola vypočítaná na základe dĺžky ozveny v pixloch vyhodnotenej modelom a jej následným prepočítaním.
- **real_begins_in_sec**: Na základe časového údaju znázorneného v ďalšom stĺpci bol zistený čas začiatku ozveny v sekundách od počiatočného okraja obrázka.
- **real_begins_in_time**: Tento údaj predstavujúci reálny čas začiatku *bounding boxu* ozveny bol zistený na základe konečnej časovej jednotky znázornenej v ďalšom stĺpci. Skladá sa z hodiny, minúty, sekundy a stotiny času v tomto poradí.
- **picture_ends_in**: Hodnoty konečných časových údajov obrázkov boli získané z názvu obrázkov. Jednotlivé snímky prípadov vstupujúce do modelu obsahujú

tieto dôležité časové údaje, ktoré sme spolu s obrázkami zaznamenali už aj v ich názve. Vďaka tomuto zisteniu je možné sledovať rôzne dôležité javy, napríklad frekvenciu výskytov jednotlivých ozvien v rôznych časoch.

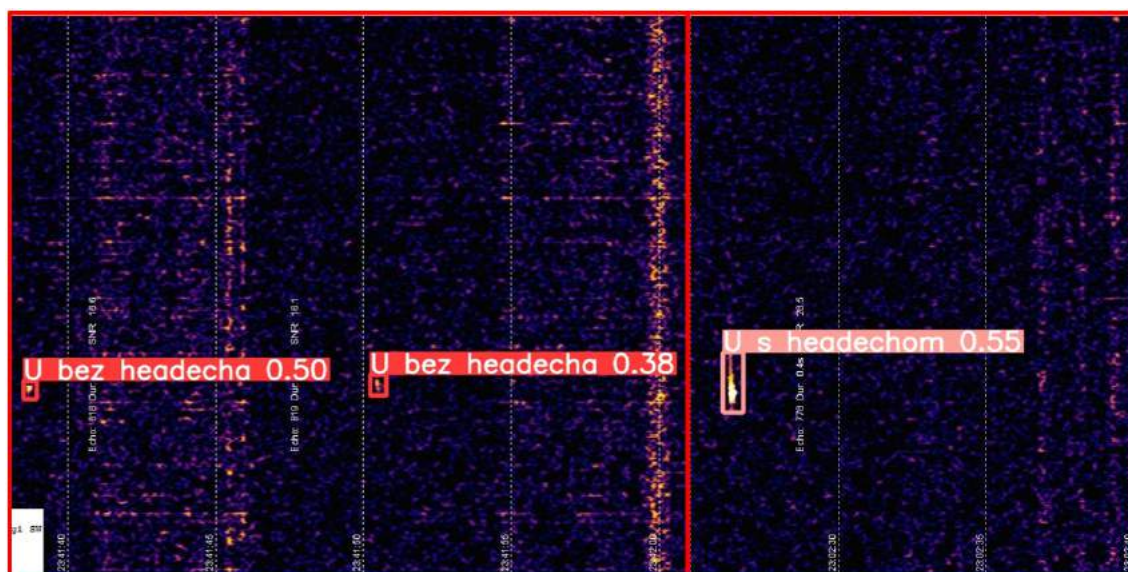
- **date:** Posledný údaj znázorňujúci dátum zhotovenia obrázka bol získaný rovnako z jeho názvu, ako aj údaj o čase. Na základe dátumových jednotiek sa môžu podobne tvoriť rôzne výstupy, napríklad v podobe grafov, ktoré by zhodnotili výskyty ozvien z pohľadu jednotlivých dní.

Túto výstupnú tabuľku bolo potrebné po jej vytvorení ešte prečistiť. Jednalo sa najmä o duplicitné označenia ozvien. Tie boli spôsobené na základe toho, že si pri niektorých ozvenách bol model istý na určité percento dvomi typmi ozveny. To znamená, že sa pri samotnej detekcii nevedel sám rozhodnúť len pre jeden druh. Obe označenia obsahovali informácie podobne, ako je znázornené na Obrázku 3–14. Pre lepšiu predstavu toho, čo predstavujú dáta z tabuľky nižšie, je možné nahliadnuť na dva príklady vyhodnotenia modelu zobrazené na Obrázku 3–15. Všetky potrebné skripty a dáta k tejto práci sú online dostupné aj na github repozitári⁴.

⁴https://github.com/kamilajencikova/meteor_detection

	event name	label	x	y	width	height	confidence	length_in_sec	real_begins_in_sec	real_begins_in_time	picture_ends_in	date
254	event20210227_011909_92	krátka bez headecha	0.36629	0.67308	0.0147783	0.0253566	0.637945	0.5341	12.0407	1:23:10.940	01:23:33	27.02.2021
256	event20210227_012333_99	krátka s headechom	0.366585	0.640254	0.0205255	0.161648	0.616688	0.7418	12.8784	1:23:10.878	01:23:33	27.02.2021
257	event20210227_012448_101	krátka bez headecha	0.368637	0.676704	0.0147783	0.0253566	0.637945	0.5341	13.0564	1:24:26.056	01:24:48	27.02.2021
258	event20210227_012534_102	krátka bez headecha	0.356732	0.675119	0.0139573	0.0253566	0.524158	0.5045	12.6409	1:25:11.640	01:25:34	27.02.2021
259	event20210227_012605_103	krátka bez headecha	0.357143	0.66561	0.0131363	0.0285261	0.571403	0.4748	12.6706	1:25:42.670	01:26:05	27.02.2021
260	event20210227_012833_105	krátka bez headecha	0.369869	0.638669	0.0205255	0.0412044	0.318841	0.7418	12.997	1:28:10.997	01:28:33	27.02.2021
261	event20210227_013008_107	krátka s headechom	0.357553	0.698891	0.0139573	0.0602219	0.554584	0.5045	12.6706	1:29:45.670	01:30:08	27.02.2021
262	event20210227_013206_108	krátka bez headecha	0.358374	0.672742	0.0155993	0.0269414	0.623523	0.5638	12.6706	1:31:43.670	01:32:06	27.02.2021
263	event20210227_013206_108	krátka bez headecha	0.799261	0.683043	0.0303777	0.0412044	0.370387	1.0979	28.3383	1:31:59.338	01:32:06	27.02.2021
264	event20210227_013222_109	krátka bez headecha	0.359606	0.683835	0.0262726	0.0396197	0.439859	0.9496	12.5223	1:31:59.522	01:32:22	27.02.2021
265	event20210227_014356_116	krátka s headechom	0.364532	0.698098	0.0147783	0.0903328	0.488469	0.5341	12.908	1:43:33.908	01:43:56	27.02.2021
266	event20210227_014518_118	krátka s headechom	0.361248	0.671949	0.0344828	0.101426	0.273861	1.2463	12.4332	1:44:55.433	01:45:18	27.02.2021
267	event20210227_014518_118	dĺhá s headechom	0.362069	0.667195	0.0361248	0.0824089	0.313051	1.3056	12.4332	1:44:55.433	01:45:18	27.02.2021
268	event20210227_014518_118	krátka bez headecha	0.511905	0.66561	0.0139573	0.0285261	0.385227	0.5045	18.2493	1:45:01.249	01:45:18	27.02.2021
269	event20210227_014518_118	krátka bez headecha	0.932266	0.695721	0.0139573	0.0348653	0.500336	0.5045	33.4421	1:45:16.442	01:45:18	27.02.2021
270	event20210227_014524_119	dĺhá s headechom	0.211412	0.666403	0.0353038	0.0776545	0.322189	1.276	7.003	1:44:56.003	01:45:24	27.02.2021
271	event20210227_014524_119	krátka bez headecha	0.361658	0.66561	0.0139573	0.0285261	0.555589	0.5045	12.819	1:45:01.819	01:45:24	27.02.2021
272	event20210227_014524_119	krátka bez headecha	0.78202	0.696513	0.0139573	0.0332805	0.518733	0.5045	28.0119	1:45:17.011	01:45:24	27.02.2021
273	event20210227_014728_124	krátka bez headecha	0.358374	0.690174	0.0139573	0.0427892	0.428893	0.5045	12.7003	1:47:05.700	01:47:28	27.02.2021
274	event20210227_014935_128	krátka s headechom	0.363301	0.666403	0.0238095	0.0554675	0.43528	0.8605	12.7003	1:49:12.700	01:49:35	27.02.2021
275	event20210227_015338_130	krátka s headechom	0.367406	0.68542	0.0188834	0.0649762	0.600977	0.6825	12.9377	1:53:15.937	01:53:38	27.02.2021
276	event20210227_020102_136	krátka bez headecha	0.353038	0.687797	0.0147783	0.0316957	0.603836	0.5341	12.4926	2:00:39.492	02:01:02	27.02.2021
277	event20210227_020102_136	krátka bez headecha	0.860016	0.693344	0.0270936	0.0332805	0.347334	0.9792	30.5935	2:00:57.593	02:01:02	27.02.2021
278	event20210227_020311_139	krátka bez headecha	0.356322	0.66561	0.0131363	0.0316957	0.375331	0.4748	12.641	2:02:48.641	02:03:11	27.02.2021
279	event20210227_020544_143	krátka bez headecha	0.364943	0.663233	0.0155993	0.0332805	0.568551	0.5638	12.908	2:05:21.908	02:05:44	27.02.2021
280	event20210227_020841_146	krátka bez headecha	0.353448	0.677496	0.0139573	0.0269414	0.490222	0.5045	12.5222	2:08:18.522	02:08:41	27.02.2021
281	event20210227_021209_147	krátka bez headecha	0.350575	0.664025	0.0164204	0.0285261	0.559998	0.5935	12.3739	2:11:46.373	02:12:09	27.02.2021
282	event20210227_022140_152	krátka s headechom	0.353448	0.689382	0.0139573	0.0507132	0.460141	0.5045	12.5222	2:21:17.522	02:21:40	27.02.2021

Obrázok 3 – 14 Časť finálnej výstupnej tabuľky po modelovaní. Zdroj: vlastné



Obrázok 3 – 15 Príklad vyhodnotenia modelom. Zdroj: vlastné

4 Záver

Na základe predloženej bakalárskej práce sme získali možnosť oboznámiť sa nie len s vytváraním modelu neurónovej siete, ale taktiež aj bližšie nahliadnuť do témy meteorov, ich zaznamenávaníu, rozdielom či radarom, ktoré sa na ich detekciu používali doposiaľ. Je pre nás veľkým prínosom pracovať na práci, ktorej idea vznikla vďaka spolupráci so Slovenskou akadémiou vied a Fakultou matematiky, fyziky a informatiky Univerzity Komenského v Bratislave.

Vďaka tejto spolupráci vzniká model umelej inteligencie, ktorá sa každým dňom vo svete zlepšuje, vyvíja a učí. Jej využitie je dnes možné pozorovať už naozaj v takmer všetkých oblastiach. Prichádza možnosť prispieť k tomuto učeníu aj vo smere vedy, ktorý doposiaľ možno nebol dostatočne rozvinutý. To nie len, že uľahčí prácu pracovníkom z Univerzity Komenského, ale jeho detailnosť môže prispieť k zlepšeniu samotnej detekcie, a to napríklad vďaka zachyteniu javov, ktoré by ľudské oko ináč nezaznamenalo. Taktiež môže byť daný model nasadený na spracovanie veľkého množstva vstupov, čo by nebolo možné len za použitia ľudských síl.

Výhodami tohto modelu je vysoká presnosť detekcií. Model sa dokázal až na zopár chýb naučiť rozoznávať to či daný objekt je, alebo nie je meteor. Dokáže určiť jeho typ z pohľadu dĺžky, taktiež z pohľadu existencie headecha. Výstup poskytuje používateľovi zhrnutie dát v podobe tabuľky, ktorá obsahuje všetky dôležité informácie o danej ozvene či už sa jedná o časovú informáciu, dátum výskytu, konkrétne parametre *bounding boxu* (slov. ohraničujúci box), či presnosť, akou si je model istý, že daný objekt je ten, ktorý detekoval.

V prípadnom pokračovaní na tejto práci je možné zlepšenie v trénovaní modelu, ktoré by prebiehalo pri lepších podmienkach výpočtového výkonu zariadenia, na ktorom by trénovanie prebiehalo. Vďaka tomu by bolo možné nastaviť vyššie hodnoty v častiach trénovacieho kódu `--batch` a `--epochs`. To by zabezpečilo detailnejšie učenie modelu, ktorý by ďalej pri testovaní dokázal lepšie rozoznávať pozitívne javy od tých falošných. Tým by sa znížili nedostatky modelu spojené práve s týmto prob-

lémom. Taktiež by bolo možné pokračovať nasadením nami vytvoreného modelu na dáta z celej európskej siete staníc, čo by slúžilo na väčšie rádiové pozorovanie meteorov.

V závere ostáva túto prácu už len nasadiť do používania, kde bude veľkým prínosom najmä vďaka tomu, že ušetrí nespočetné množstvo času, ktorý by bol rovnako ako doposiaľ využívaný prioritne na ručné prehľadávanie a spracovávanie dát. Obrovský prínos nastal aj na druhej strane, kde sa vďaka tvorbe tohto modelu autorka oboznámila s funkčnosťou, tvorbou a dôležitosťou modelov neurónových sietí. Vesmírna oblasť bola pre autorku úplne novou a neznámou oblasťou, preto bol čas venovaný tvorbe tejto práce nie len veľmi prínosný, ale aj zaujímavý.

Literatúra

- Albawi, S., Mohammed, T. A. and Al-Zawi, S. (2017). Understanding of a convolutional neural network, *2017 international conference on engineering and technology (ICET)*, Ieee, pp. 1–6.
- Aly, G. H., Marey, M. A. E.-R., El-Sayed Amin, S. and Tolba, M. F. (2021). Yolo v3 and yolo v4 for masses detection in mammograms with resnet and inception for masses classification, *International Conference on Advanced Machine Learning Technologies and Applications*, Springer, pp. 145–153.
- Ao, S.-I., Rieger, B. B. and Amouzegar, M. (2010). *Machine learning and systems engineering*, Vol. 68, Springer Science & Business Media.
- Beysolow, I. et al. (2017). Introduction to deep learning, *Introduction to Deep Learning Using R*, Springer, pp. 1–9.
- El Naqa, I. and Murphy, M. J. (2015). What is machine learning?, *machine learning in radiation oncology*, Springer, pp. 3–11.
- Fleet, R. (2015). Correlating video meteors with graves radio detections from the uk, *International Meteor Conference Mistelbach, Austria*, p. 30.
- Girshick, R. (2015). Fast r-cnn, *Proceedings of the IEEE international conference on computer vision*, pp. 1440–1448.
- Jordan, M. I. and Mitchell, T. M. (2015). Machine learning: Trends, perspectives, and prospects, *Science* **349**(6245): 255–260.
- Ko, S., Pareek, A., Ro, D. H., Lu, Y., Camp, C. L., Martin, R. K. and Krych, A. J. (2022). Artificial intelligence in orthopedics: three strategies for deep learning with orthopedic specific imaging.

- Liu, S., Qi, L., Qin, H., Shi, J. and Jia, J. (2018). Path aggregation network for instance segmentation, *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 8759–8768.
- Michalski, R. S., Carbonell, J. G. and Mitchell, T. M. (2013). *Machine learning: An artificial intelligence approach*, Springer Science & Business Media.
- Natekin, A. and Knoll, A. (2013). Gradient boosting machines, a tutorial, *Frontiers in neurorobotics* **7**: 21.
- Redmon, J., Divvala, S., Girshick, R. and Farhadi, A. (2016). You only look once: Unified, real-time object detection, *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 779–788.
- Redmon, J. and Farhadi, A. (2017). Yolo9000: better, faster, stronger, *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 7263–7271.
- Ripley, B. D. (2007). *Pattern recognition and neural networks*, Cambridge university press.
- Roopak, M., Tian, G. Y. and Chambers, J. (2019). Deep learning models for cyber security in iot networks, *2019 IEEE 9th annual computing and communication workshop and conference (CCWC)*, IEEE, pp. 0452–0457.
- Shuttleworth, S. (2016). Old weather: Citizen scientists in the 19th and 21st centuries, *Science Museum Group Journal* **3**(3).
- Sindagi, V. A. and Patel, V. M. (2018). A survey of recent advances in cnn-based single image crowd counting and density estimation, *Pattern Recognition Letters* **107**: 3–16.
- Steinwart, I. and Christmann, A. (2008). *Support vector machines*, Springer Science & Business Media.

- Thuan, D. (2021). Evolution of yolo algorithm and yolov5: the state-of-the-art object detection algorithm.
- Wang, C.-Y., Liao, H.-Y. M., Wu, Y.-H., Chen, P.-Y., Hsieh, J.-W. and Yeh, I.-H. (2020). Cspnet: A new backbone that can enhance learning capability of cnn, *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, pp. 390–391.
- Wang, S., Niu, L. and Li, N. (2018). Research on image recognition of insulators based on yolo algorithm, *2018 International Conference on Power System Technology (POWERCON)*, pp. 3871–3874.
- Wislez, J.-M. (1996). Forward scattering of radio waves off meteor trails, *Proceedings of the International Meteor Conference, 14th IMC, Brandenburg, Germany, 1995*, pp. 99–117.
- Xu, R., Lin, H., Lu, K., Cao, L. and Liu, Y. (2021). A forest fire detection system based on ensemble learning, *Forests* **12**: 217.
- Yap, M. H., Hachiuma, R., Alavi, A., Brüngel, R., Cassidy, B., Goyal, M., Zhu, H., Rückert, J., Olshansky, M., Huang, X. et al. (2021). Deep learning in diabetic foot ulcers detection: a comprehensive evaluation, *Computers in Biology and Medicine* **135**: 104596.
- Zigo, P. (2018). Kolonica-modra: the new system for forward-scatter radio meteor observations, *Proceedings of the International Meteor Conference*, pp. 76–78.

Zoznam príloh

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

Príloha B Používateľská príručka

Príloha C Systémová príručka