

Univerzita Palackého v Olomouci
Přírodovědecká fakulta
Katedra geoinformatiky

**AUTOMATIZACE IDENTIFIKACE
DISTURBANCE LESNÍCH POROSTŮ
NA ZÁKLADĚ SENTINEL DAT**

Bakalářská práce

Eliška REGENTOVÁ

Vedoucí práce: doc. RNDr. Vilém Pechanec, Ph.D.

Olomouc 2020
Geoinformatika a geografie

ANOTACE

Cílem práce je vytvoření (polo)automatizovaného nástroje pro identifikaci disturbance lesních porostů na základě Sentinel dat. Nástroj je vytvořen v programovacím jazyce Python s využitím objektově orientovaného přístupu. Vstupními daty jsou snímky z družice Sentinel 2, nad nimiž se počítají čtyři vybrané vegetační indexy – NDVI, RSR, RVSI a REIP. Vytvořený program je interaktivní a výsledné rozhodnutí o výskytu disturbance lesních porostů podporuje grafy, časovou řadou, statistickými ukazateli i barevným znázorněním hodnot indexů na satelitním snímku. Program lze využít pro jakoukoliv oblast lesa s charakterem mírného pásu. Výsledné hodnoty jsou spíše orientační pro vymezení oblastí s výraznou změnou. Přesné stanovení výskytu, příčin a následků disturbance vyžaduje další podrobnější průzkum.

KLÍČOVÁ SLOVA

Dálkový průzkum Země; Automatizace; Disturbance; Lesní porost; Sentinel 2;

Počet stran práce: 43

Počet příloh: 6 (z toho 2 volné a 4 elektronické)

ANOTATION

The aim of the thesis is creation of semi-automated tool for identification of forest disturbance based on Sentinel data. The tool is written in the Python programming language using an object-oriented approach. The input data are satellite images of Sentinel 2 on which four selected vegetation indices are calculated – NDVI, RSR, RVSI and REIP. The program is interactive and the final decision on forest disturbance is supported by graphs, time series, statistical indicators and colour representation of index values on the satellite image. The program is possible to use for any forest area with the character of a temperate zone. The results rather indicate the areas of significant change. Precise determination of the occurrence, causes and consequences of disturbance requires further detailed research.

KEYWORDS

Earth Observation; Automation; Disturbance; Forest stand; Sentinel 2;

Number of pages: 43

Number of appendixes: 6

PROHLAŠUJI, ŽE

- bakalářskou práci včetně příloh, jsem vypracovala samostatně a uvedla jsem všechny použité podklady a literaturu.
- jsem si vědoma, že na moji bakalářskou práci se plně vztahuje zákon č.121/2000 Sb. – autorský zákon, zejména § 35 – využití díla v rámci občanských a náboženských obřadů, v rámci školních představení a využití díla školního a § 60 – školní dílo,
- beru na vědomí, že Univerzita Palackého v Olomouci (dále UP Olomouc) má právo nevýdělečně, ke své vnitřní potřebě, bakalářskou práci užívat (§ 35 odst. 3),
- souhlasím, aby jeden výtisk bakalářské práce byl uložen v Knihovně UP k prezenčnímu nahlédnutí,
- souhlasím, že údaje o mé bakalářské práci budou zveřejněny ve Studijním informačním systému UP,
- v případě zájmu UP Olomouc uzavřu licenční smlouvu s oprávněním užít výsledky a výstupy mé bakalářské práce v rozsahu § 12 odst. 4 autorského zákona,
- použít výsledky a výstupy mé bakalářské práce nebo poskytnout licenci k jejímu využití mohu jen se souhlasem UP Olomouc, která je oprávněna v takovém případě ode mne požadovat přiměřený příspěvek na úhradu nákladů, které byly UP Olomouc na vytvoření díla vynaloženy (až do jejich skutečné výše).

V Olomouci dne

Eliška REGENTOVÁ

.....

Děkuji vedoucímu práce doc. RNDr. Vilému Pechancovi, Ph.D. za věcné připomínky, náměty na vylepšení, pomoc při testování aplikace i čas věnovaný vždy zajímavým a velmi přínosným konzultacím.

UNIVERZITA PALACKÉHO V OLMOUCI

Přírodovědecká fakulta

Akademický rok: 2018/2019

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(projektu, uměleckého díla, uměleckého výkonu)

Jméno a příjmení: **Bc. Eliška REGENTOVÁ**
Osobní číslo: **R17428**
Studijní program: **B1301 Geografie**
Studijní obor: **Geoinformatika a geografie**
Téma práce: **AUTOMATIZACE IDENTIFIKACE DISTURBANCE LESNÍCH POROSTŮ NA ZÁKLADĚ SENTINEL DAT**
Zadávající katedra: **Katedra geoinformatiky**

Zásady pro vypracování

Cílem bakalářské práce je tvorba nástroje pro (polo)automatizaci procesu identifikace disturbance lesních porostů. Základní funkční prvky výsledného řešení budou I) přístup k datům, II) čtení metadat, III) výpočet vhodného vegetačního indexu, IV) vyhodnocení změny (identifikace a kvantifikace) mezi dvěma časovými obdobími, V) lokalizace disturbance v prostoru. Identifikace disturbance bude založena na analýze dat z družic Sentinel 1 a 2. Nástroj bude funkční na předem zvolené a zdůvodněné platformě. Výsledným řešením práce bude funkční nástroj a základní technická dokumentace.

Celou práci, tj. text včetně všech příloh, posteru, výstupů, zdrojových i vytvořených dat, map, programových kódů a databází, student odevzdá v digitální podobě na paměťovém nosiči připevněném k deskám práce s popisem (jméno, název práce, Katedra geoinformatiky UP, rok). Text práce s přílohami odevzdá ve dvou svázaných výtiscích na sekretariát katedry ve stanoveném termínu. O práci student vytvoří webovou stránku v souladu s pravidly dostupnými na stránkách katedry. Práce bude zpracována podle obecných zásad (Voženilek, 2002) a závazné šablony pro kvalifikační práce na KGI. Povinnou přílohou práce je poster formátu A2.

Rozsah pracovní zprávy: **max. 50 stran**

Rozsah grafických prací: **dle potřeby**

Forma zpracování bakalářské práce: **tištěná**

Seznam doporučené literatury:

Metodiky a dokumenty UHUL <http://www.uhul.cz>

LUKEŠ P. et al. Hodnocení zdravotního stavu lesních porostů v České republice pomocí satelitních dat Sentinel-2. Certifikovaná metodika. UHUL. 2018
ZEMEK, F. et al. Letecký dálkový průzkum Země: teorie a příklady hodnocení terestrických ekosystémů. Brno: Centrum výzkumu globální změny AV ČR, 2014. ISBN 978-80-87902-07-3

BUCHA T. et al. Satelity ve službách lesa. Bratislava: SAP, 2014. ISBN 978-80-89607-25-9

dokumentace k programovacím jazykům a prostorovým knihovnám

program GMES/Copernicus <https://land.copernicus.eu/>

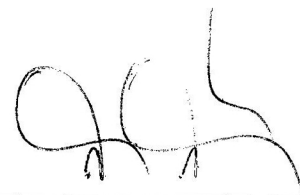
indexované databáze

Vedoucí bakalářské práce: **doc. RNDr. Vilém Pechanec, Ph.D.**

Katedra geoinformatiky

Datum zadání bakalářské práce: 6. května 2019
Termín odevzdání bakalářské práce: 6. května 2020

L.S.



doc. RNDr. Martin Kubala, Ph.D.
děkan

prof. RNDr. Vít Voženílek, CSc.
vedoucí katedry

OBSAH

SEZNAM POUŽITÝCH ZKRATEK	10
ÚVOD	11
1 CÍLE PRÁCE.....	12
2 METODY A POSTUPY ZPRACOVÁNÍ.....	13
2.1 Použité metody.....	13
2.1.1 Dálkový průzkum Země (DPZ).....	13
2.1.2 Vegetační indexy	13
2.1.3 Python.....	15
2.1.4 Objektově-orientovaný přístup	15
2.2 Použitá data	16
2.2.1 Sentinel 1.....	16
2.2.2 Sentinel 2.....	17
2.3 Použité programy	17
2.3.1 PyCharm	17
2.3.2 Anaconda.....	18
2.3.3 QGIS	18
2.3.4 Knihovny.....	18
2.4 Postup zpracování.....	18
3 SOUČASNÝ STAV ŘEŠENÉ PROBLEMATIKY	19
3.1 Identifikace disturbance lesních porostů	19
3.2 Nástroje použitelné pro identifikaci disturbance	20
4 VÝBĚR METOD A POSTUPŮ ZPRACOVÁNÍ.....	21
4.1 Vstupní data	21
4.1.1 Výhody a nevýhody použití.....	21
4.1.2 Kombinace snímků Sentinel 1 a 2	21
4.2 Výběr vegetačních indexů	22
4.3 Způsob (polo)automatizace.....	23
5 PŘÍPRAVA K TVORBĚ NÁSTROJE.....	24
5.1 Definice vstupních dat	24
5.2 Návrh grafického uživatelského rozhraní	25

5.3	Soubory pro spuštění aplikace.....	26
6	TVORBA NÁSTROJE.....	27
6.1	První okno nástroje (třída FirstPage).....	27
6.1.1	Přístup k datům.....	28
6.1.2	Čtení metadat.....	28
6.1.3	Výpočet vegetačních indexů.....	30
6.2	Druhé okno nástroje (třída SecondPage).....	32
6.2.1	Vyhodnocení změny.....	32
6.2.2	Lokalizace změny v prostoru.....	33
7	VÝSLEDKY.....	34
7.1	Výsledky testování výpočtů a maskování.....	34
7.2	Soubor o průběhu výpočtu a tabulka změn.....	34
7.3	Program FDI_tool.....	35
8	DISKUZE.....	37
9	ZÁVĚR.....	38
	POUŽITÁ LITERATURA A INFORMAČNÍ ZDROJE	
	PŘÍLOHY	

SEZNAM POUŽITÝCH ZKRATEK

BSD	Berkeley Software Distribution
CSV	Comma-separated Values
ČVUT	České vysoké učení technické v Praze
DPZ	Dálkový průzkum Země
ESA	European Space Agency
EU/ICP Forests	International Cooperative Programme on Assessment and Monitoring of Air Pollution Effects on Forests
GIS	Geographic Information System
GML	Geography Markup Language
GPL	General Public License
GPS	Global Positioning System
GRD	Ground Range Detected
JP2	Joint Picture Experts Group 2000
LAI	Leaf Area Index
MIR	Mid-Infrared Spectroscopy
MIT	Massachusetts Institute of Technology License
MSI	Moisture Stress Index
NDII	Normalized Difference Infrared Index
NDVI	Normalized Difference Vegetation Index
NIR	Near Infrared Spectroscopy
OOP	objektově orientované programování
PyPI	Python Package Index
RED	červené absorpční pásmo
REIP	Red Edge Inflection Point
ROI	Region-of-Interest
RSR	Reduced Simple Ratio
RVSI	Red-edge Vegetation Stress Index
SAR	Synthetic Aperture Radar
SHP	Shapefile
SLC	Single Look Complex
TIFF	Tagged Image File Format
UAV	Unmanned Aerial Vehicle
ÚHÚL	Ústav pro hospodářskou úpravu lesů
UTM	Universal Transverse Mercator coordinate system
VI	Vegetační index
WGS84	World Geodetic System 1984
WKB	Well-Known Binary
WKT	Well-known text
XML	Extensible Markup Language
YAML	YAML Ain't Markup Language

ÚVOD

Dálkový průzkum Země (DPZ) je neustále se rozvíjející metoda snímání zemského povrchu. Nabízí produkty s vysokým prostorovým rozlišením v několika spektrálních pásmech, které lze vhodnou kombinací použít k vyhodnocení parametrů v různých oborech. Jedním z mnoha takových oborů je i lesnictví. Zhoršující se zdravotní stav lesních porostů lze pomocí DPZ indikovat dříve, než je tato změna vůbec patrná lidským okem. Data se sbírají v jeden okamžik pro velká území, díky čemuž jsou mezi sebou lépe porovnatelná. Vzhledem k jejich dostupnosti i bezplatnému použití se jeví čas strávený inovací stávajících metod vyhodnocujících zdravotní stav lesních porostů jako přínosný.

Automatizace procesů je trendem dnešní doby. Usnadňuje lidem práci a její využití lze nalézt téměř ve všech oborech lidské činnosti. Investice do automatizace, ať už se jedná o průmyslovou výrobu, služby nebo vědu se vyplatí. I vyhodnocení zdravotního stavu lesních porostů lze automatizovat. Vhodnou kombinací těchto dvou metod vznikne nástroj, který dokáže identifikovat místa s pozitivní i negativní změnou ve stanoveném časovém horizontu. Správná automatizace zkrátí čas nutný ke sběru dat i k jejich pozdějšímu vyhodnocování ve statistických programech. Pro vytvoření automatického nástroje je třeba vybrat takové parametry rostlin, které lze na snímcích z DPZ identifikovat a které mají dostatečnou vypovídající hodnotu o jejich zdravotním stavu. Od nástroje se očekává jednoduchost v používání z pohledu uživatele a dostatečná interaktivita umožňující rozhodování příslušníků lesní správy, ekologů, environmentalistů, vědců či dalších zájemců o tuto problematiku.

1 CÍLE PRÁCE

Cílem této bakalářské práce je tvorba nástroje pro (polo)automatizaci procesu identifikace disturbance lesních porostů. Základní funkční prvky výsledného řešení budou I) přístup k datům, II) čtení metadat, III) výpočet vhodného vegetačního indexu, IV) vyhodnocení změny (identifikace a kvantifikace) mezi dvěma časovými obdobími, V) lokalizace disturbance v prostoru. Identifikace disturbance bude založena na analýze dat z družic Sentinel 1 a 2. Nástroj bude funkční na předem zvolené a zdůvodněné platformě. Výsledným řešením práce bude funkční nástroj a základní technická dokumentace.

Teoretická část je rozdělena do dvou logických částí. První z nich se zabývá běžně používanými způsoby monitorování zdravotního stavu vegetace, rozebírá pojem „disturbance“ lesních porostů a představuje problematiku spektrální odrazivosti vegetace. Druhá část se zabývá současnými dostupnými softwarovými řešeními, která lze využít pro výzkum zdravotního stavu lesních porostů.

Praktická část popisuje vytvořený (polo)automatizovaný nástroj a funkcionalitu, která je stanovena v zadání této bakalářské práce. Součástí je obhajoba výběru vstupních dat a technologií. Je zde rovněž vysvětlen proces výběru vegetačních indexů, jež jsou zakomponovány do výsledného nástroje. Následující kapitoly rozebírají samotný postup tvorby aplikace včetně ukázek zdrojového kódu.

Hlavním přínosem této práce je využití veřejných dat z družic Sentinel pro identifikaci disturbance lesních porostů, čímž lze výrazně snížit náklady na zjišťování jejich zdravotního stavu v dlouhodobém časovém horizontu. Disturbance lesů navíc může být analyzována častěji, než tomu bylo doposud. Dále řešení práce využívá graf časové řady a umožňuje zobrazit i výsledné předchozí snímky oblasti. Nevýhodou navrženého způsobu měření je však závislost na atmosférických podmínkách – zejména oblačnosti. Tento problém je ale možné vyřešit tzv. mozaikováním, kdy se z více snímků pořízeným v krátkém časovém období poskládá mozaika z jejich částí bez oblačnosti. Tento způsob korekce dat z optického snímkování je blíže popsán v diskuzi, jelikož není předmětem této bakalářské práce.

2 METODY A POSTUPY ZPRACOVÁNÍ

Metody, data, programy a postupy zpracování použité pro tvorbu nástroje k identifikaci disturbance lesních porostů jsou stručně představeny v následujících podkapitolách. Způsob implementace do vlastního aplikačního řešení se nachází v praktické části.

2.1 Použité metody

Tato bakalářská práce se opírá o čtyři základní metody a přístupy. Data jsou získávána distančně – dálkovým průzkumem Země. Hodnocení zdravotního stavu vegetace probíhá na základě výpočtů vegetačních indexů. Aplikace je postavena na principu objektově orientovaného programování v jazyce Python.

2.1.1 Dálkový průzkum Země (DPZ)

Dálkový průzkum Země je v Terminologickém slovníku zeměměřičství a katastru nemovitostí definován jako „sběr dat o území realizovaný z kosmického (kosmický průzkum) nebo letadlového nosiče (letecký průzkum) a zpracování těchto dat k získání informací o poloze, stavu a druhu objektů a jevů zemského povrchu“ (Terminologický slovník VÚGTK). DPZ se v praxi mimo jiné využívá k distančnímu šetření zdravotního stavu vegetace na základě jejich vlastností spektrální odrazivosti. Hlavní výhodou použití této metody oproti terénnímu měření je možnost získání dat z velkého území v rámci velmi krátkého časového intervalu.

V případě disturbance lesních porostů je výhoda plošného sběru dat v krátkém časovém období stěžejní vzhledem k rozloze lesů – dle Ústavu pro hospodářskou úpravu lesů (ÚHÚL) dosahoval podíl lesů na celkové rozloze České republiky v roce 2018 celkem 34,1 %, tj. 2,67 mil. ha (Katalog rychlých informací, ÚHÚL), a také vzhledem k měnícím se spektrálním odrazivostem vegetace v důsledku životních projevů rostlin, tzv. fenologických fází.

2.1.2 Vegetační indexy

Vegetačním indexem (VI) se dle skript ČVUT (Kolář, 1997) rozumí rozdíl odrazivosti naměřených v blízkém infračerveném spektrálním pásmu (NIR) a v absorpčním pásmu v červené barvě (RED). Prudký nárůst hodnot mezi těmito pásmy je charakteristický pro vegetaci a indikuje přítomnost zelené hmoty. Obecné matematické vyjádření vegetačního indexu je následující:

$$VI = NIR - RED \quad (\text{Vzorec 1})$$

V rámci tvorby nástroje, který je cílem této bakalářské práce, byly využity čtyři vybrané indexy. Prvním z nich je normovaný diferenční index (NDVI), nabývající hodnot v intervalu $<-1; 1>$ a měřící zdraví vegetace právě na základě výše uvedeného rozdílu odrazivosti v NIR a RED (Kolář, 1997).

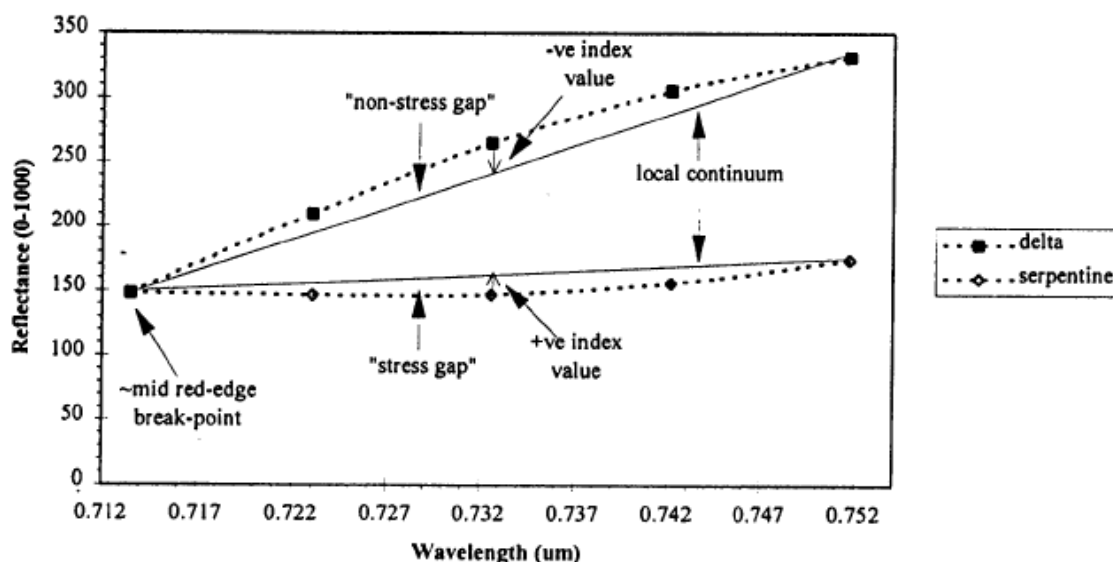
$$NDVI = \frac{(NIR - RED)}{(NIR + RED)} \quad (\text{Vzorec 2})$$

Druhým indexem je Reduced Simple Ratio (RSR), který je citlivý na množství nadzemní biomasy. Často je používán jako alternativa k indexu listové plochy neboli indexu pokryvnosti listoví (LAI). Hodnoty se pohybují v intervalu <0; 10>. Výpočet kombinuje kromě odrazivosti NIR a RED také střední infračervené pásmo (MIR) (Lukeš, 2018; Majasalmi, 2016).

$$RSR(LAI) = \frac{NIR}{RED} * MIR_{max} - \frac{MIR}{MIR_{max}} - MIR_{min} \quad (\text{Vzorec 3})$$

Třetí index Red-edge Vegetation Stress Index (RVSI) posuzuje stres vegetace podle odvozování míry spektrální konkávnosti, která je definována jako posun odrazivosti v datech od lineárně modelovaného středního bodu (střední odrazivosti) nacházejícího se mezi bodem červeného okraje spektrální křivky a začátkem blízkého infračerveného pásma. Dle míry konkávnosti jsou výsledné hodnoty kladné nebo záporné (Merton, 1998).

$$RVSI = \left(\frac{718 \text{ nm} + 748 \text{ nm}}{2} \right) - 733 \text{ nm} \quad (\text{Vzorec 4})$$



Obr. 1: Grafické znázornění RVSI (zdroj: Merton, 1998)

Posledním indikátorem používaným v této bakalářské práci je pozice inflexního bodu červeného okraje spektrální křivky (REIP). Posun tohoto bodu směrem do delších vlnových délek je způsoben vlivem rozšíření červeného pásma z důvodu vyššího obsahu chlorofylu ve zkoumané vegetaci (Cho & Skidmore, 2006; Atzberger, 2008).

$$REIP = 700 + 40 \left(\frac{\left(\frac{670 \text{ nm} + 780 \text{ nm}}{2} \right) - 700 \text{ nm}}{740 \text{ nm} - 700 \text{ nm}} \right) \quad (\text{Vzorec 5})$$

2.1.3 Python

Python je multiplatformní skriptovací programovací jazyk rozvíjený od roku 1991 pod open source licencí Python Software Foundation Licence, která jej dělá volně použitelným a distribuovatelným, dokonce i pro komerční použití. Python Package Index (PyPI) zprostředkovává tisíce Python modulů třetích stran. Instalační balíček pro Python je dodáván se standardní knihovnou a dalšími komunitně rozvíjenými moduly. Postupně vznikly tři vzájemně nekompatibilní verze – Python 1, Python 2 a v současné době používaný Python 3 (*python.org*).

Jazyk Python podporuje více paradigmat. Umožňuje programování v procedurálním, objektivě orientovaném, funkcionálním stylu i v jejich kombinaci (*Summerfield, 2010*). Mezi hlavní charakteristiky jazyka patří dynamická kontrola datových typů, což znamená, že teprve při běhu programu (ne jeho kompilaci) se proměnným deklaruji příslušné datové typy podle obsahu. Jakmile jsou však proměnné jednou definovány, nedochází k jejich automatickému přetypování (*ITnetwork: Python*). Python je citlivý na použití velkých a malých písmen, tzv. case-sensitive. Další výhodou je významná podpora k integraci s ostatními jazyky a nástroji (*py.cz*).

Základními datovými typy jsou Booleovské operátory 0 = False, 1 = True, numerické datové typy int pro celá čísla, float pro desetinná čísla psaná s desetinnou tečkou, sekvenční typy list, tuple a range, dále také textový datový typ str, binární bytes a bytearray nebo slovníkový typ dict, ale i mnoho dalších. Výčet všech dostupných datových typů je uveden v oficiální dokumentaci (*python.org*).

Z dotazníkového šetření a analýzy dat webu Stack Overflow za rok 2019 je Python v rámci této IT komunity čtvrtým nejpoužívanějším programovacím jazykem, a to u 41,7 % respondentů. Celkem 25,7 % uvedlo přání se Python naučit a 73,1 % jej označilo jako svůj nejoblíbenější programovací jazyk (*Stack Overflow*).

2.1.4 Objektově-orientovaný přístup

Objektově orientované programování (OOP) je dle přístup k vývoji softwaru, kterou podporuje většina programovacích jazyků. V jeho implementaci je kladen důraz především na znovu použitelnost jednotlivých komponent. Základní jednotkou je objekt, který představuje předmět zájmu z reálného světa. Tento objekt je charakterizován vlastnostmi neboli atributy. Dále je schopen vykonávat činnosti, které se v OOP označují jako metody. Vzor, podle kterého je instance objektu vytvořena se nazývá třída. Třída má definovány atributy a metody, které jsou společné pro všechny její instance.

OOP je postaveno na třech základních pilířích, jimiž jsou zapouzdření, dědičnost a polymorfismus. Zapouzdření rozděluje metody na veřejně přístupné a vnitřní, použitelné jen v rámci dané třídy. Dědičnost je schopnost třídy dědit atributy a metody z hierarchicky nadřazené třídy. Podtřídě je možné doplnit nové atributy a metody nebo upravit zděděné, přičemž tyto změny na nadřazenou třídu nemají vliv. Polymorfismus naopak vytváří jednotné rozhraní pro práce s různými typy objektů. Všechny tyto objekty např. mohou mít společnou metodu, která je ale u každého typu objektu vykonávána jiným způsobem (*ITnetwork: Návod*).

2.2 Použitá data

V rámci evropského programu Copernicus jsou provozovány čtyři řady družic Sentinel (1, 2, 3 a 5), poslední řada s označením Sentinel 4 je zatím stále ve vývoji. První tři řady jsou v konstelaci dvou družic A, B. Veškerá data jsou dostupná zdarma ve službě Copernicus Open Access Hub (scihub.copernicus.eu).

Mise Sentinel 1 funguje na principu radarového snímání a je zaměřena na monitorování pevniny, moří, oceánských vod, pobřeží, ledu i polárních oblastí. V České republice je však využíván v oblasti krizového řízení – zejména při povodních, díky vysoké rychlosti dostupnosti dat po jejich pořízení.

Mise Sentinel 2 poskytuje multispektrální snímky určené především pro monitorování krajinného pokryvu a územních změn. Senzor těchto družic zahrnuje tři pásma v červeném okraji viditelného spektra, díky čemuž je vhodný především pro zkoumání stavu vegetace. Pro analýzy snímků z družic Sentinel 2 se používají vegetační indexy, uvedené v kap. 2.1.2 a 4.2. Nevýhodou je případné zastínění snímané oblasti vysokou mírou oblačnosti.

2.2.1 Sentinel 1

První družice s označením Sentinel 1A odstartovala 3. dubna 2014 o dva roky později bylo vypuštěna i její párová dvojice Sentinel 1B. Zemi obíhají po stejné dráze ve výšce 693 km, se vzájemným posuvem o 180°. Každá družice nasnímá Zemi nad rovníkem 175krát za 12 dní. Obě jsou nosičem senzoru C-Band SAR (Synthetic Aperture Radar) s jedním frekvenčním pásmem (5 404 MHz). Prostorové rozlišení závisí na režimu snímání a pohybuje se v rozmezí 5–40 m na pixel.

V jednom snímku je schopen zabrat 20–400 km. Družice Sentinelu 1 jsou schopny vysílat i přijímat série impulzů ve vertikální i horizontální polarizaci. Polarizační schémata jsou zvolena s ohledem na snímkanou oblast – pro polární, mořské a zaledněné oblasti se jedná o HH-HV nebo HH polarizaci, pro ostatní oblasti VV-VH nebo VV polarizace. Nabízí 4 režimy snímání:

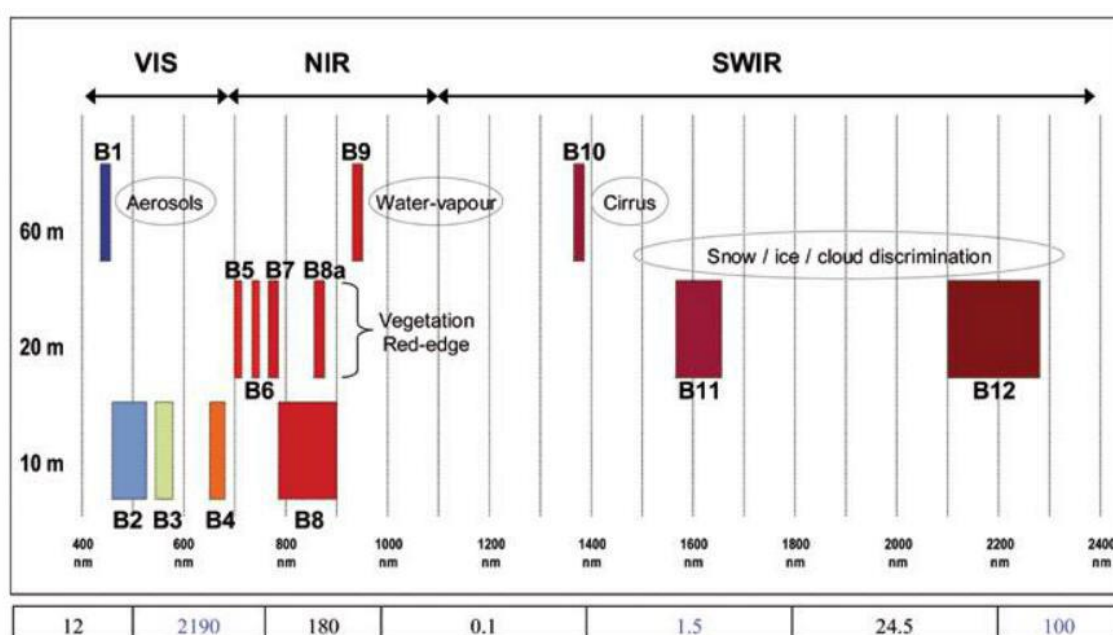
- Strip Map Mode s rozlišením 5 m na pixel a šířkou 80 km
- Interferometric Mode s rozlišením 5 × 20 m a šířkou záběru 250 km
- Extra Wide Mode s rozlišením 20 × 40 m a šířkou 400 km
- Wave Mode s rozlišením 5 × 5 m a šířkou záběru 20 x 20 km

Snímky z družic Sentinel 1 jsou distribuovány ve třech úrovních zpracování. L0 produkty se skládají pouze z komprimovaných surových dat. Obsahují informaci o poloze na oběžné dráze a vnitřní kalibraci, ale i šum. Úroveň L1 označuje data vzniklá z L0, na nichž byly provedeny geometrické transformace obrazu. Single Look Complex (SLC) úroveň L1 označuje data georeferencovaná na základě informací o výšce a poloze družice. Ground Range Detected (GRD) data jsou georeferencována podle referenčního zemského elipsoidu, a navíc jsou zbaveny termálního šumu, což přispívá k vyšší kvalitě snímku. Data úroveň L2 jsou obohaceny o další geofyzikální komponenty za účelem sledování vln, mořských proudů nebo větru (*CollGS: Sentinel 1*).

2.2.2 Sentinel 2

Mise Sentinel 2 snímá obdobně jako její předchůdce – Sentinel 1. Dvě družice snímají opět na společné dráze ve výšce 786 km ve vzájemném posunu o 180°. Časové rozlišení snímků je však nižší, každá družice získá data o daném území jednou za 10 dnů. Sentinel 2A je v provozu od roku 2015, druhá družice, Sentinel 2B, byla vypuštěna o dva roky později.

Optický multispektrální senzor MSI dokáže zaznamenat vlnové délky odpovídající rozsahu od viditelné části (443 nm) po střední infračervenou část spektra (2 190 nm). Tento záběr dělí do 13 pásem v prostorovém rozlišení 10, 20 a 60 m na pixel. Data Sentinel 2 lze získat ve třech úrovních zpracování dat. L1B obsahuje radiometrické korekce i korekce geometrie senzoru. L1C je ortorektifikovaná v projekci UTM / WGS84. Úroveň L2A lze dosáhnout invencí uživatele s využitím Sentinel 2 toolboxu (*CollGS: Sentinel 2*).



Obr. 2: Spektrální pásma družice Sentinel 2 (zdroj: *CollGS: Sentinel 2*)

2.3 Použité programy

Pro samotné programování aplikace bylo použito vývojové prostředí PyCharm. K dosažení celkové funkcionality přispěly moduly i externí knihovny se svobodnou licencí. Správnost výpočtů a maskování snímku byla ověřena v geoinformačním systému QGIS.

2.3.1 PyCharm

Ačkoliv je PyCharm převážně komerční software od české vývojářské společnosti JetBrains, nabízí i verzi PyCharm Community Edition spadající pod svobodnou Apache Licence. PyCharm poskytuje podporu pro nástroje Django i Anaconda (*JetBrains*). Řešení této bakalářské práce bylo zpracováno ve verzi 2019.2.2.

2.3.2 Anaconda

Anaconda je platforma pod New BSD Licence pro zprostředkování jazyka Python a R pro oblast data science, machine learning i data processing. Ve svém uživatelském prostředí Anaconda Navigator usnadňuje práci s instalací jednotlivých modulů a knihoven a zajišťuje kompatibilitu jejich verzí (*Anaconda Cloud*). Instalace v rámci této bakalářské práce probíhala skrze verzi Anaconda 3.

2.3.3 QGIS

Výsledky výpočtů a maskování satelitních snímků byly ověřeny v geoinformačním systému QGIS, který je distribuován pod licenci GPL (*QGIS*). Použita byla starší verze QGIS Desktop 3.8.3.

2.3.4 Knihovny

PyQt 5.9.2 je set vazeb toolkitu Qt pro jazyk Python. Qt toolkit je aplikační rámec napsaný v jazyce C++, ve kterém je obsažena řada komponent včetně těch pro vývoj uživatelského prostředí. PyQt navíc přináší přes 35 modulů speciálně pro Python. Práva užívání a další distribuce tohoto setu jsou ošetřena licenci GPL v3 (*PyQt*).

Rasterio 1.1.0 nabízí Python API pro čtení, zápis a operace s rastrovými daty založené na n-rozměrných homogenních polích NumPy a formátu GeoJSON. Je uvolněný pod BSD licenci (*Rasterio*).

PyQtGraph 0.10.0 představuje knihovnu pro vývoj grafického uživatelského prostředí Python aplikací. Zaměřuje se na vizualizaci dat a interakci uživatele s nimi. Je distribuován pod MIT licenci (*PyQtGraph*).

Fiona 1.18.11 je zaměřená na čtení a zápis datových souborů včetně vícevrstevých vektorových formátů používaných v oblasti geoinformatiky. Do řešení bakalářské práce byla vybrána pro jednoduchost a podporu Esri Shapefile formátu. Taktéž je užívána pod licenci BSD (*Fiona*).

Rasterstats 0.14.0 je modul pro výpočet statistických hodnot geoprostorových dat v rastrovém datasetu v rámci vektorové vrstvy. Podporuje GeoJSON, WKT / WKB geometrii i datové formáty podporované v API Fiona (*Rasterstats*).

Dále byly pro běžné funkce, práci se složkami a parsování XML dokumentu použity moduly integrované v instalačním balíčku Python 3.7.4 – **os**, **sys**, **fnmatch** a **xml.etree.ElementTree**.

2.4 Postup zpracování

Postup zpracování programu byl obdobou fází životního cyklu dle J. Poláka. Úvodní studie představovala rozbor zadání práce, sběr požadavků, stahování cvičných dat Sentinel 1 a 2 a rešerši v oblasti vzájemně kombinovatelných technologií pod svobodnou licenci. Byly posouzeny konkurenční nástroje, jejich výhody i nevýhody. Následovala analýza, která určovala přesné vlastnosti a funkcionalitu výsledné aplikace. Po důsledném zvážení těchto vstupních parametrů byl vytvořen návrh grafického prostředí a vytyčily se jednotlivé programovatelné bloky. Poslední fází bylo programování a propojování jednotlivých komponent a funkcí v každé této definované sekci. Po jejím dokončení bylo provedeno testování a oprava chyb v dané části programu (*Polák, 2003*).

3 SOUČASNÝ STAV ŘEŠENÉ PROBLEMATIKY

„Disturbance neboli narušování, je spojeno s přímou destrukcí vytvořené biomasy“ (Míchal, 1994). „Představuje významné a často náhlé změny v ekosystémech“ (Křižová, Ujházy, 2007). Mezi hlavní příčiny poškození lesa patří invaze lesních škůdců (podkorního nebo dřevokazného hmyzu, hub), introdukce invazivních druhů, sucho, znečištění ovzduší, acidifikace, eroze půd, klimatická změna, požáry nebo větrná kalamita (Neumann et al., 2017; Millar et al., 2015). Přírozené disturbance určují dynamiku lesních ekosystémů, jejich důsledkem je vytvoření nového prostředí, mají nezastupitelnou roli ve formování prostorové struktury lesa i při zachování jeho biologické rozmanitosti (Havira et al., 2017).

Disturbance se v DPZ projevuje změnou spektrální odrazivosti vegetace. Pro její správnou identifikaci se sledují biochemické, strukturní, fenologické či morfologické parametry rostlin. Nejčastěji se ze senzoru získávají data o množství vody v pletivech, obsahu pigmentů, podíl fotosynteticky aktivního záření absorbovaného rostlinou, biomasa nebo druhové složení lesa (Lausch et al., 2016). Životní projevy rostlin se mění v závislosti na okolních podmínkách i v průběhu roku. Studium těchto periodických změn se zabývá fenologie. Celé období se dělí do tzv. fenologických fází, kdy v každé z nich jsou projevy rostlin jiné. Je tedy důležité porovnávat snímky pořízené ve stejné fenologické fázi, neboť v opačném případě může dojít k výraznému zkreslení a mylné identifikaci disturbance (Fenologie, 2020).

3.1 Identifikace disturbance lesních porostů

V současné době se pro identifikaci disturbance lesních porostů v České republice využívá analýza dat z terénního měření nebo z kombinace distančního a terénního šetření. Při využití distančního způsobu sběru dat dochází v první fázi ke snímkování oblasti pomocí letadel nebo UAV zařízení. Jeho výhodou je získání dat pro další analýzy rychle, s vysokým rozlišením a za velmi krátké období, čímž jsou data vzájemně porovnatelná. Nevýhodou je možný vliv atmosféry, nutnost vlastnit snímací zařízení a vysoké provozní náklady. Po analýze těchto dat se na místech s pravděpodobným výskytem disturbance sbírají podrobnější data získaná destruktivní metodou – odběrem vzorků z kmene – přímo v terénu. Poté následuje opětovné zpracování získaných dat statistickými metodami.

V případě, že distanční způsob sběru dat není v identifikaci disturbance lesních porostů využit, rozdělí se oblast na segmenty bez okolních vlivů. V těchto přibližně stejně velkých segmentech se geodeticky zaměřují živé i odumřelé stromy a ležící kmeny, vizuálně se posuzuje stupeň rozkladu, vypočítává se stupeň pokryvnosti a zapisují se jednotlivé parametry stromů (výška, obvod kmene, hustota zápoje apod.). Geodetická data se zpracovávají nejčastěji v programu FieldMap, pro výpočty statistických ukazatelů je využit jazyk R nebo program Microsoft Excel (Zenáhlíková, 2012; Havira et al., 2017).

Při zpracování satelitních dat bývá použita nejčastěji metodika vyvinutá společností Stoklasa Tech v rámci projektu International Cooperative Programme on Assessment and Monitoring of Air Pollution Effects on Forests (EU/ICP Forests). Zpracovává snímky z družice Landsat 7 a vegetační indexy MSI a NDII (Lukeš, 2018). Sami autoři však uvádí obtížnou až nemožnou verifikaci výsledků z důvodu nízkého rozlišení senzoru a s tím souvisejícího obtížného zaměřování kontrolních ploch (Stoklasa, Fabiánek, 1998).

Samotný projekt EU/ICP Forests navíc hodnotí zdravotní stav lesů v Evropě na úrovni jednotlivých stromů na základě řady binárních parametrů (s odpovědí ano / ne), což může negativně ovlivnit přesnost hodnocení (*ICP Forests*).

3.2 Nástroje použitelné pro identifikaci disturbance

Na trhu je v současnosti dostupná řada nástrojů pro zpracování dat získaných z DPZ. Mezi tyto nástroje patří SNAP od Evropské kosmické agentury (ESA), Envi od společnosti Harris Geospatial Solutions, ERDAS Imagine vyvíjený firmou Hexagon Spatial nebo také mnohá GIS řešení pracující obecně s rasterovými daty – ArcGIS od Esri, open-source nástroj QGIS, GRASS GIS i knihovna GDAL. Jejich velkou předností je množství nabízených funkcí, rychlost, stabilita i technická podpora. Některé z nich jsou dostupné pod open-source licencí, jiné jsou poměrně nákladné na pořízení. Všechny uvedené ale vyžadují schopnost uživatele s nimi pracovat nebo mít základy programování. Časová osa disturbancecí lesních porostů je dostupná v Envi a částečně i ve nástroji SNAP skrze možnost zobrazení více snímků najednou. Grafy je často nutné vygenerovat samostatně. Řešení této bakalářské práce je postaveno na velmi jednoduchém řešení bez nutnosti dalšího programování. Na uživatele jsou kladeny minimální požadavky z oborů IT a GIS.

4 VÝBĚR METOD A POSTUPŮ ZPRACOVÁNÍ

Data, metody a postupy zpracování uplatněné v této bakalářské práci byly zvoleny na základě mnoha faktorů. Jejich stěžejní výhody, ale i důležité limity jsou popsány v následujících podkapitolách.

4.1 Vstupní data

Kapitola Vstupní data je rozdělena do dvou částí – rozboru klíčových pozitiv a omezení a popisu metody kombinace satelitních snímků z družic Sentinel 1 a 2. Spojování snímků za účelem zvýšení přesnosti dat však bylo z důvodu časové náročnosti programu a nižší míry zpřesnění zejména u lesních ploch nakonec vypuštěno ze samotného řešení práce.

4.1.1 Výhody a nevýhody použití

Hlavní výhodou misí Sentinel je veřejně dostupný bezplatný archiv v portálu Copernicus Open Access Hub. Po zvolení příslušné lokality, družice a období sběru dat je k dostání velké množství snímků v dostatečném časovém i prostorovém rozlišení pro výzkum zdravotního stavu lesů. Dostatek spektrálních pásem nabízí možnost použití mnoha indexů s vysokou citlivostí na pozorované charakteristiky vegetace. Atmosférické korekce jsou ve snímcích s vyšší úrovní zpracování již zahrnuty (*CollGS: Sentinel 1, Sentinel 2*).

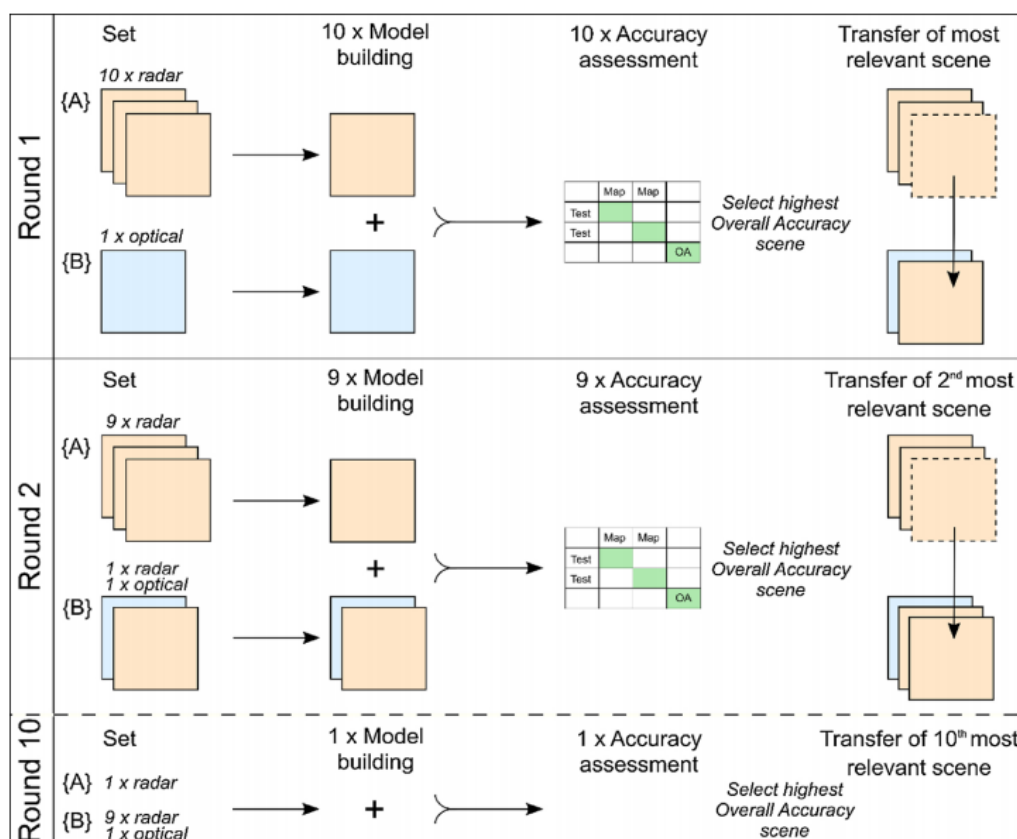
Na rozdíl od jiných metod dálkového průzkumu Země jsou družicová data mise Sentinel z hlediska nákladů vhodnější pro „pravidelný plošný monitoring aktuálních stavů i dlouhodobých trendů zdravotního stavu vegetace“ (*Jones, Vauhan, 2010; Rautianen et al., 2010*). Jejich použití je limitováno dobou snímání teprve od roku 2014 a 2015. U dat z družice Sentinel 2 navíc závisí na překryvu území oblačností.

4.1.2 Kombinace snímků Sentinel 1 a 2

Kombinace snímků družic Sentinel 1 a 2 při výpočtu vegetačních indexů dle studie M. J. Steinhausena et al. (2018) prokazatelně zvyšuje přesnost výsledků. Pro zpracování dat touto metodou je nutné nejprve provést atmosférické a geometrické korekce snímků, což lze usnadnit použitím snímků Sentinel 2 na úrovni zpracování L2A a Sentinel 1 úrovně GRD s aplikací algoritmů Range Doppler Terrain Correction a LeeSigma, jejichž výstupem je georeferencovaný radarový snímek se shlazenou texturou „soli a peře“.

Samotná technika zpřesnění vstupních dat probíhá pomocí modelu Random Forest, kam vstupuje vždy jeden snímek optického senzoru a několik radarových snímků (*Obr. 3*). Inkluze snímku Sentinel 2 s nejrelevantnějším radarovým snímkem družice Sentinel 1 dle výše uvedené studie zvýšila míru celkové přesnosti dat o 2,92 % na pixel. Stálý a výrazný nárůst celkové přesnosti je pak viditelný do kombinace jednoho snímku Sentinel 2 se čtyřmi snímků Sentinel 1 (*Steinhausen et al., 2018*).

Vzhledem k hlavnímu přínosu vytvořeného nástroje – výpočtu vegetačních indexů v časové řadě a možnosti posouzení disturbance z několika snímků, byl nakonec pro zvýšení efektivity proces inkluze snímků ze dvou družic vypuštěn. Významnou roli v tomto rozhodnutí hrál výpočetní čas programu, ale i nízké procento nárůstu přesnosti v poměru ke zvýšení vytížení paměti i procesoru v případě kombinace dat. Možnost rozvoje vytvořeného nástroje je dále popsána v diskuzi (*kap. 8*).



Obr. 3: Postup kombinace snímků Sentinel 1 a 2 (zdroj: Steinhäusen et al., 2018)

4.2 Výběr vegetačních indexů

Výběr vegetačních indexů byl inspirován odbornou literaturou týkající se problematiky hodnocení zdravotního stavu vegetace (Zemek et al., 2014; Bucha et al., 2014; Albrechtová et al., 2017; Merton, 1998; Lukeš, 2018), příp. přímo lesních porostů. Bylo nutné brát v úvahu typ vegetace i její prostředí, ve kterém se nachází. Odrazivost vegetace se mění rovněž v závislosti na její fenologické fázi v době snímání (Zhang et al., 2003). Fenologii rostlin se zabývá například studie E. Clelandové (2007), v níž je zmíněna především citlivost vegetace na sezónnost prostředí a posuny v načasování činnosti rostlin. Pro správné vyhodnocení indexů a určení míry disturbance v časové řadě je tedy potřeba dbát na volbu vstupních dat v jedné fenologické fázi.

Nejčastěji používaným indexem pro vegetaci je normalizovaný diferenční vegetační index (NDVI), citlivý na množství biomasy. Pro podporu této významné charakteristiky byl zvolen ještě Reduced Simple Ratio (RSR). Podstatnou vlastností vegetace je i obsah vody v pletivech, jež je hodnocen např. na základě pozice inflexního bodu červeného okraje spektrální křivky (REIP). V dnešní době vegetaci ovlivňují také okolní vlivy – znečištění půdy, ovzduší, klimatická změna, napadení škůdci nebo fyzicko-geografické jevy (sucho, mráz, sesuvy, větrné kalamity apod.). Tyto vlivy mají dopad na míru stresu vegetace, kterou hodnotí Red-edge Vegetation Stress Index (RVSI).

Proměnné vybraných vegetačních indexů musely být před samotnou aplikací vzorců převedeny do spektrálních pásem družice Sentinel 2. Tento satelit disponuje 13 pásmy, jejichž šířka a střední vlnová délka je popsána v datové specifikaci mise Sentinel 2 (*CollGS: Sentinel 2*). Odpovídající pásmo bylo stanoveno právě na základě této dokumentace. Převedené vzorce vypadají následovně:

$$NDVI = \frac{B08 - B04}{B08 + B04}$$

(Vzorec 6)

$$RSR = \left(\frac{B07}{B04}\right) * \frac{B11_{max} - B11}{B11_{max} - B11_{min}}$$

(Vzorec 8)

$$RVSI = \left(\frac{B05 + B06}{2}\right) - B06_{min}$$

(Vzorec 7)

$$REIP = 700 + 40 * \frac{\left(\frac{B04 + B07}{2} - B05\right)}{B06 - B05}$$

(Vzorec 9)

4.3 Způsob (polo)automatizace

Předpokládanou cílovou skupinou nástroje jsou odborníci z řad environmentalistů, lesní správy nebo ochránců přírody. Nástroj slouží pro analýzu zdravotního stavu vegetace a rozhodování o disturbanci lesů mírného pásu nebo lesů podobného složení. Pro interpretaci výsledků je vyžadována odborná znalost fyziologického stavu lesních porostů a vlivů na něj působících. Po této cílové skupině není požadováno geoinformatické vzdělání, ani zkušenosti s programováním či skriptováním. Z tohoto důvodu byla zvolena tvorba samostatného programu s jednoduchým uživatelským prostředím, zaměřeného na konkrétní analýzy, namísto skriptů, které by doplnily stávající produkty určené k práci se satelitními snímky.

5 PŘÍPRAVA K TVORBĚ NÁSTROJE

Praktickou částí této bakalářské práce je vytvoření (polo)automatizovaného nástroje pro identifikaci disturbance lesních porostů na základě Sentinel dat. Programovací a skriptovací jazyk Python, ve kterém byl nástroj napsán, byl vybrán s ohledem na podporu objektivně orientovaného přístupu, jednoduchost syntaxe i na četnost využití v geoinformatické praxi. U jednotlivých knihoven, API a modulů se posuzovala kompatibilita, podporované vstupní i výstupní formáty, a především nabízené funkce.

5.1 Definice vstupních dat

Před započítím programování bylo nutné definovat jednotlivé komponenty, jejich stěžejní funkce, příslušnost ke třídě, vazby a dědičnost objektů. Řada těchto definic byla převzata z již existujících knihoven – např. PyQt5 a PyQtGraph, ke kterým dané objekty patří. Bylo ale potřeba vytvořit i nové.

Pro popis satelitních dat družic Sentinel byla stanovena abstraktní třída `SatelliteImage`. Její stěžejní funkcí, v případě dalšího vývoje programu (např. implementace výpočtů pro Sentinel 1, maskování oblačnosti apod.), je zabránění redundance dat pro snímky družic Sentinel 1 a 2. Tyto snímky mají společné vlastnosti – název, soubor s metadaty, satelit, datum a čas snímání a souřadnice v XML gramatice GML.

```
class SatelliteImage(object):
    images = [] # pole všech objektů této třídy

    def __init__(self, name):
        self.name = name # parametr třídy SatelliteImage
        self.md_file = "" # zbylé atributy se doplňují později
        self.satellite = ""
        self.date = ""
        self.time = ""
        self.gml_coordinates = ""
        self.images.append(self) # přidání objektu do pole images
```

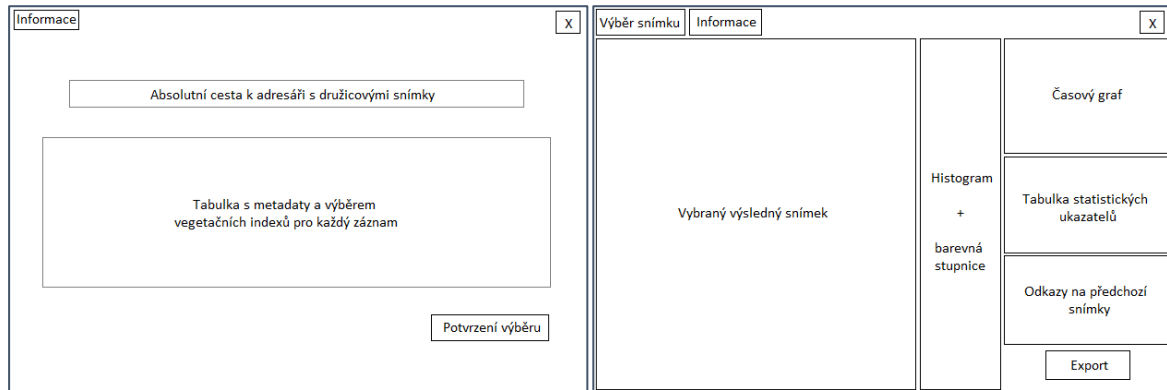
Z abstraktní třídy `SatelliteImage` dědí všechny vlastnosti instance podtřídy `S2`. Tato podtřída je obohacena o další vlastnosti a metody, které jsou specifické právě pro snímky družic Sentinel 2. Je to zejména procento pokryvu snímku oblačností, úroveň zpracování a identifikátor dlaždice. Z metod je to rozbor souboru s metadaty, výběr a načtení spektrálních pásem, určení rozlišení snímku a výpočty vegetačních indexů.

```
class S2(SatelliteImage):
    images = [] # pole objektů třídy S2
    dirname = "" # cesta ke složce se snímky S2

    def __init__(self, name):
        super().__init__(name) # objekty přebírají parametr name
        self.cloud_cover = "" # rozšíření atributů snímků třídy S2
        self.processing_level = ""
        self.ndvi = False # výpočty VI jsou defaultně zakázány
```


5.2 Návrh grafického uživatelského rozhraní

Program se skládá ze dvou klíčových částí – načtení snímků, metadat a výběru vegetačních indexů a vizualizační interaktivní části, zobrazující výsledky výpočtů a změnu disturbance v prostoru. Proto bylo navrženo rozdělení programu na dvě samostatná okna.



Obr. 4: Návrh grafického uživatelského rozhraní (zdroj: Autorka BP, 2020)

Ve zdrojovém kódu programu byla tato okna definována obdobně jako vstupní data – pomocí abstraktní třídy. Třída Page je charakterizována velikostí okna a vlastním menu se záložkou About, kde se nachází základní informace o programu. Tato třída dědí vlastnosti a metody PyQt komponenty s názvem QMainWindow. Z třídy Page dědí dvě podtřídy FirstPage a SecondPage, jejichž metody jsou blíže popsány v kap. 6.

```
class Page(QMainWindow): # třída Page dědí z QMainWindow

    def __init__(self):
        super().__init__()
        self.resize(800, 600) # změna defaultní velikosti okna
        self.setWindowTitle("FDI_tool")

    # použití widgetů z knihovny PyQt - QMenuBar pro menu aplikace
    # a QDialog pro záložku About o programu:

    self.menubar = QtWidgets.QMenuBar(self)
    self.menubar.setGeometry(QtCore.QRect(0, 0, 800, 20))
    self.menubar.setObjectName("menubar")
    self.setMenuBar(self.menubar)

    def infoAbout(self):
        win = QtWidgets.QDialog(self)
        win.setWindowTitle("About")
        win.setGeometry(QtCore.QRect(300, 100, 600, 370))
```

5.3 Soubory pro spuštění aplikace

Součástí programu musí být také soubory potřebné ke spuštění aplikace. Klíčový soubor je nazván Main. Skrze něj probíhá prvotní spuštění, nastavení proměnných prostředí a volání metod doplňujícího souboru Control, v němž se nachází metody pro dynamické vytvoření instancí třídy S2, chybových hlášek, načtení satelitního snímku i zavření prvního a otevření druhého okna aplikace.

```
# vytvoření proměnné prostředí aplikace s cestou k GDAL souboru proj.db
os.environ['PROJ_LIB'] = "${CONDA_PREFIX}/Library/share/proj"

# spuštění aplikace

def main():
    app = QtWidgets.QApplication(sys.argv)
    win = FirstPage()
    win.show()
    sys.exit(app.exec_())

# volání metod pro vytvoření objektů třídy S2 a výpočet hodnot VI

def call():
    Control.getImage(S2)
    S2.countValues(self)

if __name__ == "__main__":
    main()
```

6 TVORBA NÁSTROJE

V kapitole Tvorba nástroje jsou popsána řešení dílčích cílů práce ze zdrojového kódu. Jedná se o přístup k datům, čtení metadat, výpočet vegetačních indexů, vyhodnocení změny a lokalizace této změny v prostoru. Kapitola rovněž přibližuje obsah tříd FirstPage a SecondPage, které se věnují vizuální stránce aplikace a jejichž návrh je představen v kap. 5.2. Používání aplikace z pohledu uživatele je rozebráno v příložené elektronické dokumentaci ve formátu PDF i na webových stránkách této bakalářské práce.

6.1 První okno nástroje (třída FirstPage)

Úkolem prvního okna je poskytnout uživateli metadata pro možnost volby vstupních dat, vegetačních indexů a ořezových masek. V souboru FirstPage.py je využita knihovna PyQt5 pro vytvoření grafického uživatelského prostředí prvního okna nástroje. Dále bylo potřeba provázat třídu s dalšími soubory aplikace – abstraktní nadřazenou třídou Page v souboru Page.py, třídou S2, kde se nachází definice dat z družic Sentinel 2, soubor Control.py, pro řízení chodu aplikace a volání stěžejních funkcí, a nakonec s druhým oknem SecondPage, jenž se otevírá na konci této programové části. Funkcionalitu prvního okna nástroje zajišťují také integrované moduly jazyka Python os, sys a io. Třída FirstPage má jeden vlastní atribut data_in_table, který nabývá hodnoty True po vložení absolutní cesty k adresáři s extrahovanými satelitními snímky Sentinel 2 a stisknutí tlačítka Load.

```
class FirstPage(Page): # třída FirstPage dědí ze třídy Page

    def __init__(self):
        super().__init__()
        self.data_in_table = False
        self.setFixedSize(self.size())
        self.setupUi()
```

Tlačítka, adresní řádek i tabulka v prvním okně aplikace představují widgety knihovny PyQt5, konkrétně pak modulu QtWidgets. Tlačítka Browse, Load a OK převzaly základní vlastnosti z třídy QPushButton, adresní řádek pochází z widgetu QLineEdit a tabulka (QTableWidget) je vytvořena z jednotlivých řádků a sloupců dědicích ze třídy QTableWidgetItem. Tabulka je rovněž zasazena do oblasti s možností interaktivního vertikálního i horizontálního posouvání widgetu QScrollArea.

```
self.lineEdit = QtWidgets.QLineEdit(self)
self.lineEdit.setGeometry(QtCore.QRect(70, 125, 360, 20))
self.lineEdit.setObjectName("lineEdit")

self.Browse = QtWidgets.QPushButton(self)
self.Browse.setGeometry(QtCore.QRect(484, 120, 80, 30))
self.Browse.setText("Browse")
self.Browse.setObjectName("Browse")
self.Browse.clicked.connect(self.browseDirs) # aktivace tlačítka

self.scrollArea = QtWidgets.QScrollArea(self)
self.scrollArea.setGeometry(QtCore.QRect(20, 170, 761, 271))
self.scrollArea.setWidgetResizable(True)
self.scrollArea.setObjectName("scrollArea")
```

6.1.1 Přístup k datům

Po stisknutí tlačítka Browse se automaticky volá metoda `browseDirs()`, jež uživateli umožňuje procházet složky ve svém počítači. Těto funkcionality je dosaženo pomocí modulu `QWidgets` a prvku `QFileDialog`. Získání adresy ke složce s extrahovanými satelitními snímky, staženými z Copernicus Open Access Hubu zajišťuje třída `QDir` v modulu `QtCore`. Touto adresou je naplněn jediný atribut, `dirname`, třídy `Control`, a její text je rovněž zapsán metodou `setText()` do adresního řádku v aplikaci.

```
def browseDirs(self):
    if self.lineEdit.text() == "":
        dirname = QtWidgets.QFileDialog.getExistingDirectory(self,
            "Select data directory", QtCore.QDir.rootPath())
        dirname = QtCore.QDir.toNativeSeparators(dirname)
    if os.path.isdir(dirname):
        self.lineEdit.setText(dirname)
        Control.dirname = dirname
```

Z absolutní cesty definované uživatelem se následně načtou všechny dostupné snímky družice Sentinel 2. Každý snímek splňující podmínku koncovky `SAFE` a názvu satelitu `S2` je instancí třídy `S2`. Pro každou tuto instanci se zjišťují potřebná metadata.

```
@classmethod
def createInstances(cls):
    # metoda třídy, v případě dat S1 se
    # pouze doplní poslední podmínka if
    items = os.listdir(Control.dirname)
    print("Number of files: " + str(len(items)))
    for item in items:
        if item.endswith(".SAFE"):
            if item.startswith("S2"):
                S2_image = S2(item)
                S2.getMetaDataFile(S2_image)
```

6.1.2 Čtení metadat

Čtení metadat je druhou požadovanou funkcionalitou nástroje dle zadání této bakalářské práce. Metadata snímku se nachází v XML souboru v oficiální složce s daty. Jeho název začíná písmeny `MTD`. Úkolem metody `getMetaDataFile()` je najít právě takový soubor, který splňuje podmínku názvu a umístění. Jestliže jej nalezne, vytáhne z jeho struktury informace o satelitu, datu a čase snímání, souřadnicích, pokryvnosti snímku oblačností, úrovni zpracování a identifikátoru dlaždice snímaného území.

```
def getMetaDataFile(self):
    for path, sub_dirs, files in os.walk(self.dirname + "/" +
        self.name):
        for file in files:
            # nalezení XML souboru s metadaty
            while file.startswith("MTD"):
                self.md_file = file
                self.parseXML(file)
            return
```

```

def parseXML(self, file):          # metoda parsování XML dokumentu
    print(self.dirname + " " + self.name + " " + file)
    tree = Etree.parse(self.dirname + "/" + self.name + "/" + file)
    root = tree.getroot()
    self.satellite = root[0][0][9][0].text
    self.date = formatDateTime(root[0][0][9][2].text, 0)
    self.time = formatDateTime(root[0][0][9][2].text, 1)
    self.gml_coordinates = root[1][0][0][0][0].text
    self.cloud_cover = "{:.3f}".format(float(root[3][0].text))
    self.processing_level = root[0][0][3].text
    self.tile_id = self.name[39:44]

```

Získané informace o satelitním snímku se zapisují do tabulky v prvním okně aplikace. Podmínka v metodě createTable() přidává do každé buňky požadovaný text nebo funkci. Například do každého řádku ve sloupci odpovídajícím vegetačním indexům přidá zaškrťovací okénko, na základě kterého uživatel určí, zda se daný index pro příslušný snímek vypočítá či nikoliv.

```

def createTable(self):
    num_rows = len(S2.images)      # dynamické načtení záznamů
    self.tableWidget.setRowCount(num_rows)
    for x in range(0, num_rows):
        self.tableWidget.setVerticalHeaderItem(x,
        QtWidgets.QTableWidgetItem())
        for y in range(0, 11):     # vytvoření a definice buněk tabulky
            if y < 6:
                attr = self.switch_cols(y)
                self.tableWidget.setItem(x, y,
                QtWidgets.QTableWidgetItem(getattr(S2.images[x], attr)))
            elif y == 10:
                input_shp = QtWidgets.QTableWidgetItem()
                input_shp.setFlags(QtCore.Qt.ItemIsSelectable)
                self.tableWidget.setItem(x, y, input_shp)
            else:
                checkbox = QtWidgets.QTableWidgetItem()
                checkbox.setFlags(QtCore.Qt.ItemIsUserCheckable
                | QtCore.Qt.ItemIsEnabled)
                checkbox.setCheckState(QtCore.Qt.Unchecked)
                self.tableWidget.setItem(x, y, checkbox)
    self.tableWidget.resizeColumnsToContents()
    self.tableWidget.itemClicked.connect(self.handleItemClicked)

```

Při zaškrtnutí nebo odznačení okénka pro volbu vegetačního indexu se vyplní logický atribut u objektu (vybraného snímku) třídy S2. V případě vložení masky k ořezání snímku ve formátu Shapefile se uloží i absolutní cesta k tomuto souboru. Do tabulky se načítá libovolný počet snímků. Je však třeba brát v úvahu, že čím je množství dat určených k dalším výpočtům a analýze vyšší, tím je program pomalejší.

6.1.3 Výpočet vegetačních indexů

Výpočet vybraných vegetačních indexů probíhá na pozadí aplikace a pro jeho dokončení není potřeba uživatelské invence. Proces se spouští automaticky se stisknutím tlačítka OK v prvním okně aplikace. Skládá se z pěti klíčových metod – `getBands()`, `getBand()`, `countValues()`, `createOutDir()` a `extractByMask()`.

První část algoritmu vybere potřebná pásma vylučovací metodou. Soustava podmínek je sestavena tak, aby ve výsledku zbyla pouze ta pásma, která jsou potřeba k výpočtu všech zvolených vegetačních indexů. Voláním metody `getBand()` pro každé pásmo z výsledné množiny zajistí prohledání složky daného snímku a uložení adresy ke konkrétnímu spektrálnímu pásmu ve formátu JP2 s hodnotou prostorového rozlišení dle oficiální dokumentace k datům Sentinel 2.

```
def getBands(self):
    bands = ["B04", "B05", "B06", "B07", "B08", "B11"]
    if not self.ndvi: # postupné odebrání spektr. pásem
        bands.remove("B08")
        if not (self.rsr or self.reip):
            bands.remove("B04")
    if not self.reip:
        if not self.rsr:
            bands.remove("B07")
        if not self.rvsi:
            bands.remove("B05")
            bands.remove("B06")
    if not self.rsr:
        bands.remove("B11")

    if bands:
        print("Retrieving of spectral bands for the image "
              + self.name[0:60])
        for band in bands:
            result = self.getBand(band)
            if result is None:
                print(band + " image was not found.")
                return
            else:
                print(result)
    else:
        print("No spectral bands needed for the image "
              + self.name[0:60])
```

Metoda `countValues()` následně dle atributů daného snímku postupně volá metody pro výpočty jednotlivých vegetačních indexů `countNDVI()`, `countRSR()`, `countREIP()` a `countRVSI()`, jež jsou založeny na stejné struktuře. V první části se ověří, zdali již výsledný snímek neexistuje nebo zda je potřeba vytvořit ořezovou masku. Tato sekce končí nejpozději vypsáním stavové hlášky „Preparing data...“ v případě, že daný snímek zatím nebyl vytvořen. Dalších pět řádků se věnuje načítáním spektrálních pásem. Jestliže je třeba načíst pásma různého rozlišení, převedou se všechna tato pásma bilineární interpolací na pásma s nižším rozlišením. Následuje samotný výpočet podle oficiálního

vzorci. Výsledek se zapíše do souboru ve formátu TIFF se stejným souřadnicovým systémem jako mají vstupní data. V případě vyplnění atributu shp_path adresou k vektorové vrstvě dojde k ořezání výsledného snímku.

```
def countNDVI(self):
    if os.path.isfile(self.dirname + "/Output_data/" + self.name[7:26]
        + self.name[37:44] + "_10m_NDVI.tiff"):
        print("File " + self.name[7:26] + self.name[37:44]
            + "_10m_NDVI.tiff already exists.")
    if self.shp_path != "":
        shp_name = ((os.path.basename(self.shp_path)).split(".")[0]
            if os.path.isfile(self.dirname + "/Output_data/"
                + self.name[0:27] + shp_name + "_10m_NDVI.tiff"):
            print("File " + self.name[0:27] + shp_name
                + "_10m_NDVI.tiff already exists.")
            return
        else:
            self.extractByMask("_10m_NDVI.tiff")
            return
    else:
        return
else:
    print("Preparing data...")

with rasterio.open(self.B04_10, driver='JP2OpenJPEG') as b4:
    B04 = b4.read()
with rasterio.open(self.B08_10, driver='JP2OpenJPEG') as b8:
    B08 = b8.read()
print("Bands for calculating NDVI have been loaded.")

ndvi_formula = (B08.astype(float) - B04.astype(float))
                / (B08.astype(float) + B04.astype(float))

with rasterio.open(self.dirname + "/Output_data/" + self.name[7:26]
    + self.name[37:44] + "_10m_NDVI.tiff", "w", driver="Gtiff",
    width=b4.width, height=b4.height, count=1, crs=b4.crs,
    transform=b4.transform, dtype=rasterio.float32)
    as output_image:
        output_image.write(ndvi_formula.astype(rasterio.float32))

if self.shp_path != "":
    self.extractByMask("_10m_NDVI.tiff")
return
```

Důležité jsou i metody createOutDir() pro vytvoření složky Output_data v zadané složce s daty, kde se uloží všechna výstupní data a extractByMask() pro ořezání vstupního snímku zadanou vektorovou vrstvou v SHP. Společně s výpočtem se také zavře první okno a otevře se druhé okno nástroje.

6.2 Druhé okno nástroje (třída SecondPage)

Druhé okno je interaktivnější část programu. Jeho úkolem je přiblížit změny hodnot indexů, ukázat rozložení těchto hodnot v prostoru a podat ucelené informace o zadaném území. Uživatel v něm může prohlížet dílčí zájmové oblasti pomocí polygonu ROI (z angl. Region-of-Interest). Snímky jednoho území jsou seřazeny sestupně dle data snímání a jejich průměrné hodnoty zobrazeného indexu se zobrazují ve sloupcovém grafu s časovou řadou. Tato časová řada spolu se statistickými údaji a odkazy na jednotlivé snímky v časové řadě je hlavním přínosem této bakalářské práce, neboť srovnáním indexů v čase zatím nedisponuje žádný ze svobodně dostupných softwarových řešení. Hodnoty statistických ukazatelů pro ROI ve všech snímcích lze exportovat do formátu CSV. Pro vytvoření grafického uživatelského prostředí byly použity komponenty knihovny PyQtGraph. Funkcionalitu zajišťují moduly os, rasterio, fiona, fmatch a rasterstats.

Snímky se načítají z dynamického menu pomocí metody `openImage()`. Tato metoda vybere nejnovější snímek dle zvoleného vegetačního indexu a názvu oblasti. Snímek je načten ve formátu TIFF v 32bitové hloubce typu float. Se změnou barevné stupnice se jeho kvalita nemění, neboť barevný překryv je pouze maskou nad původním snímkem dle hodnot pixelů na snímku. Spolu s načteným snímkem se aktualizuje graf časové osy, odkazy pod ním, barevná stupnice i rozsah hodnot na histogramu dle zvoleného indexu (metodou `setHistRange()`).

```
def setHistRange(self, vi):
    colors = [[0, 0, 0, 255], [80, 0, 255, 255], [220, 0, 0, 255],
              [250, 245, 30, 255], [40, 230, 40, 255],
              [30, 150, 40, 255], [30, 150, 40, 255]]

    if vi == "NDVI":
        self.histogram.setLevels(-1, 1)
        self.histogram.setHistogramRange(-1.0, 1.0, 0.1)
        self.histogram.item.gradient.setColorMap(pyqtgraph.ColorMap(
            ([0.0, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0], colors))
```

6.2.1 Vyhodnocení změny

Vyhodnocení změny probíhá výpočtem zonální statistiky pro ROI polygon. Výsledkem jsou vypočtené základní statistické ukazatele pro všechny snímky v časové řadě exportované do tabulky ve formátu CSV. Takto lze porovnat minimální i maximální hodnotu indexu vymezené oblasti, medián, průměr, statistickou odchylku i počet pixelů který daný ROI polygon pokrývá. Geometrie ROI polygonu se ukládá pro možnost dalšího použití do validního souboru GeoJSON s automaticky přiřazeným identifikátorem v názvu souboru. V tomto kroku nastal problém se souřadnicovým systémem snímku. Satelitní data družice Sentinel 2 jsou specifikována souřadnicovým systémem WGS84 v automaticky přidělované zóně UTM dle snímané oblasti v metrech a ve vektoru v tomto systému probíhá i výpočet zonální statistiky knihovnou rasterstats. GeoJSON naopak vyžaduje systém WGS84 používaný v datech GPS ve formátu desetinných stupňů. Transformace mezi těmito systémy standardně probíhá použitím knihovny GDAL. Nicméně, současná verze knihovny není v tomto kroku příliš stabilní. V průběhu může nastat chyba s načtením souboru PROJ.db. Z tohoto důvodu jsou souřadnice ROI polygonu raději ukládány do dvou GeoJSON souborů – `geojson` pro uložení validního souboru a `dočasný geojson_zs` pro výpočet zonální statistiky. Tento problém je vyřešen

v metodách `createGeoJSONtoExport()` a `createGeoJSONforZS()`. Pro opětovné načtení GeoJSON souboru do programu je vytvořena také CSV tabulka s názvem `coords_32633`, ve které je sloupec ID, odpovídající názvu souboru GeoJSON, a také sloupec se souřadnicemi lomových bodů v souřadnicovém referenčním systému WGS84 v požadované zóně projekce UTM.

6.2.2 Lokalizace změny v prostoru

Lokalizace změny v prostoru je naplněna prostřednictvím náhledu snímku s vypočítaným vegetačním indexem. Zjišťování oblastí s výskytem výrazných změn usnadňuje barevná hypsometrie, histogram rozložení hodnot pixelů snímku, graf časové řady a odkazy s možností prokliku na všechny dostupné předchozí snímky pro danou lokalitu a vegetační index. Funkcionalita komponent je převzata z knihovny `PyQtGraph` a upravena. V pravém horním rohu komponenty snímku se zobrazuje hodnota indexu konkrétního pixelu. Tuto funkcionalitu zajišťuje metoda `mouseMoved()` díky sledování pohybu kurzoru. Výsledné hodnoty indexů jsou zaokrouhleny na tři desetinná místa.

```
def mouseMoved(self, ev):
    if self.graphicsPlot.sceneBoundingRect().contains(ev)
        and self.img.image is not None:
        mousePoint = self.img.getViewBox().mapToItem(self.img, ev)
        if 0 <= mousePoint.x() <= len(self.data[0])
            and 0 <= mousePoint.y() <= len(self.data):
            text = ""
            if self.data[int(mousePoint.y())][int(mousePoint.x())]
                == -999.0:
                text = "Pixel value = null"
            else:
                text = "Pixel value = %0.3f"
                % (self.data[int(mousePoint.y())][int(mousePoint.x())])
            self.pixelValue.setText(text)
```

Pro graf časové řady byla založena třída `PlotWidget`, založená na komponentě `PlotWidget` v knihovně `PyQtGraph`. Charakterizují ji atributy `x` a `y`, nesoucí zobrazovaná data a možnost automatického roztažení dat v grafu. Třída je vybavena metodami `updateGraph()`, `drawGraph()` a `clearGraph()`. Tyto metody mají za úkol aktualizaci dat v grafu jejich vykreslení, vyčištění a zaokrouhlení taktéž na tři desetinná místa.

7 VÝSLEDKY

Veškerá výstupní data se zapisují do složky `Output_data`, která je vytvořena uvnitř uživatelem zadaného vstupního adresáře. Uvnitř se nachází výstupní snímky, textový soubor se stavovými hláškami o průběhu výpočtu, CSV soubor s tabulkou změny hodnot VI a podsložka `GeoJSON` s validními soubory formátu `GeoJSON` a CSV souborem pro uchování souřadnic lomových bodů ROI polygonu pro opětovné nahrání do aplikace.

7.1 Výsledky testování výpočtů a maskování

Výpočet vegetačních indexů ve výsledném programu vytvoří do složky `Output_data` soubory ve formátu `TIFF`, a to s názvem dle úrovně zpracování dat, datem a časem snímání, vymezenou oblastí, prostorovým rozlišením a příslušným vegetačním indexem. V případě ořezání snímku maskou ve formátu `SHP` je vytvořen celkový snímek i ořezaný. Důvodem je omezení knihovny `rasterstats`, která poskytuje příslušnou metodu v jednom procesu pouze pro jedno pásmo. Výsledné snímky byly vypočteny také v programu `QGIS` pro porovnání kvality provedení z hlediska případného snížení bitové hloubky obrazu během výpočtu. Snímky byly naprosto totožné a zachovaly si svou 32bitovou hloubku. Jakákoliv chybějící data – např. právě z důvodu ořezu snímku – byla stanovena na extrémní hodnotu `-999`. Při maskování může nastat chyba knihovny `GDAL`, která v současné době zatím není ze strany poskytovatele ošetřena. Tato chyba se zobrazuje v příkazovém řádku a poukazuje na neschopnost otevřít formát `TIFF`. Dle diskuzního fóra na `GitHubu` je možným řešením použití vícevláknového paralelního zpracování procesu (*Issue 1686, GitHub*). Tato varianta se nicméně stále testuje a nebyla do nástroje, který je předmětem této bakalářské práce zahrnuta.

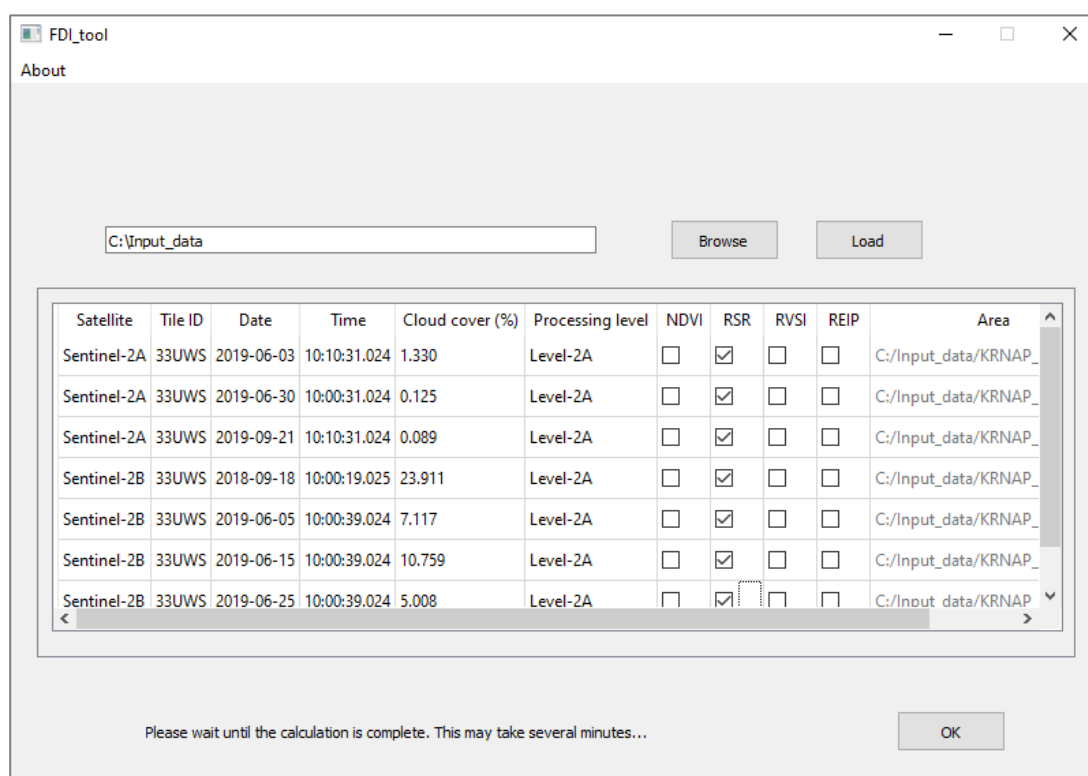
7.2 Soubor o průběhu výpočtu a tabulka změn

V souboru `ResultMessage.txt` jsou vypsány stavové hlášky o průběhu získávání spektrálních pásem, validaci názvu souboru vytvořeného souboru v zadaném adresáři a výsledku procesu všech výpočtu vegetačních indexů. Výpočet se není proveden v případě, že není dohledáno potřebné spektrální pásmo, nastala chyba při tvorbě složky výstupních dat (`Output_data`) nebo pokud již existuje soubor s názvem, který připadá nově vypočtenému snímku. Stavové hlášky z interaktivních částí nejsou do tohoto textového souboru vypsány, jelikož jsou kromě dvou popsanych chyb knihovny `GDAL` odladěny chybovými hláškami a upozorněními. Uživatel je tak schopen reagovat okamžitě.

Po stisknutí tlačítka `Export` v druhém okně aplikace se vygeneruje tabulka změn ve formátu `CSV`. Soubor se nachází ve složce `Output_data` a nese název `export.csv`. Vytváří se při prvním exportu statistických ukazatelů a následně po každém jeho smazání. Hodnoty se v něm s dalším vygenerováním nepřepisují, ale zapisují se pod sebe. Každá sekce začíná geometrií exportovaného ROI polygonu, a dále obsahuje uživatelskou poznámku (název oblasti), identifikátor příslušného souboru `GeoJSON`, název vegetačního indexu a statistické ukazatele – minimální i maximální hodnotu, průměr, medián, směrodatnou odchylku a počet pixelů, spadajících do výpočtu. V sekci jsou hodnoty všech těchto ukazatelů pro každý dostupný snímek charakterizovaný příslušným datem a změnou oproti předešlému měření. Při otevření v programu `MS Excel` si uživatel může vytvořit pro tyto hodnoty graf, díky kterému snáze vizuálně odhadne případnou disturbanci lesních porostů.

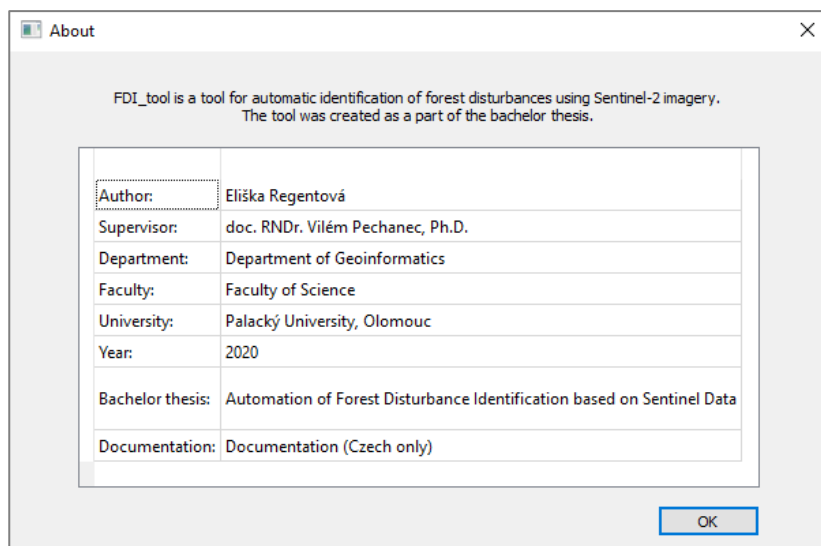
7.3 Program FDI_tool

Podrobný popis výsledného programu včetně možných chyb a limitů je uložen na CD přiloženém k této bakalářské práci, na webových stránkách i na GitHubu¹. FDI_tool byl vytvořen v jazyce Python 3.7.4 pro operační systém Windows. Jeho spuštění vyžaduje několik knihoven. Pro jejich správu je doporučen software se svobodnou licencí Anaconda 3. Seznam potřebných knihoven, modulů a balíčků je vypsán ve vygenerovaném souboru **FDI_venv.yml**. Samotné spuštění se provádí skrze příkazový řádek cmd.exe. První okno programu FDI_tool slouží k uvedení cesty adresáře s extrahovanými satelitními snímky, staženými z oficiálního archivu Copernicus Open Access Hub, které vstoupí do výpočtů. Okno se skládá z menu, adresního řádku, tlačítek a dynamicky načítané tabulky. Menu obsahuje záložku About, kde jsou uvedeny metadata o programu.



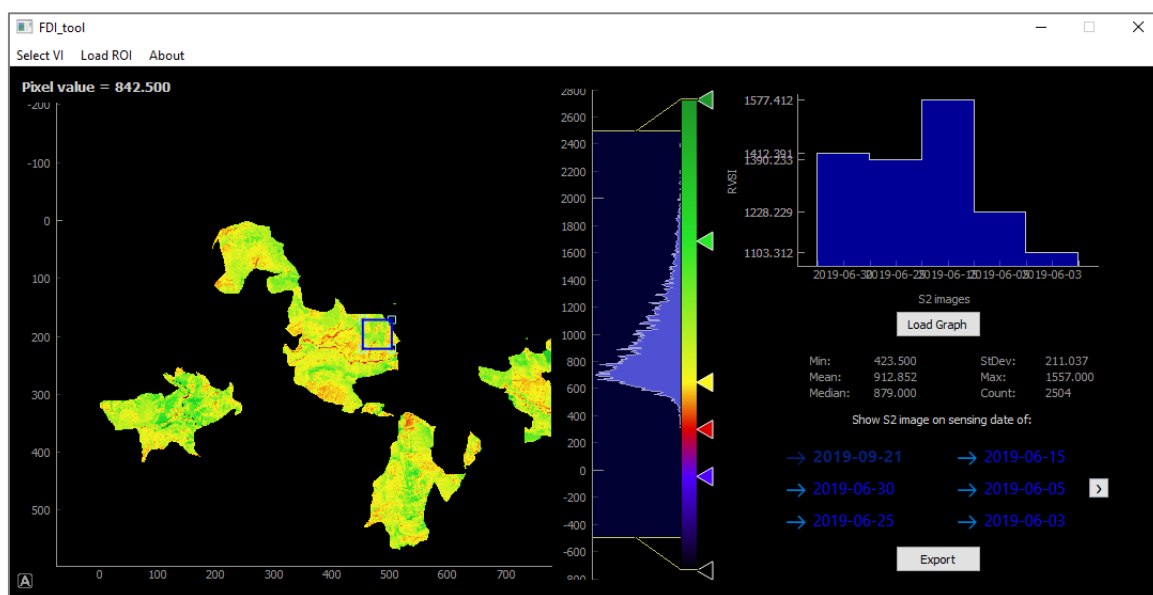
Obr. 5: První okno programu FDI_tool (zdroj: Autorka BP, 2020)

¹ Odkaz na GitHub: https://github.com/eliskareg/FDI_tool.git



Obr. 6: Metadata k programu (zdroj: Autorka BP, 2020)

Druhé okno aplikace slouží k analýze hodnot vypočtených vegetačních indexů. Je vytvořeno z dynamicky generovaného menu a několika interaktivních komponent blíže popsanych v kapitole 6.2 a v elektronické dokumentaci. Snímek je načítán skrze záložku Select VI. Po jeho zobrazení se automaticky přednastaví barevná stupnice. Tuto stupnici lze libovolně měnit – upravovat barvy, posouvat, přidávat nebo odebírat barevné záložky. Oblast snímku je možné libovolně přibližovat, oddalovat nebo posouvat klasickým pohybem myši nebo pomocí touchpadu. V levém horním rohu oblasti prohlížení se nachází hodnota vegetačního indexu konkrétního pixelu odečítaná při najetí počítačovou myši do jeho oblasti. Modrý čtvercový polygon zobrazený v oblasti snímku definuje uživatelem zvolený ROI (Region-of-Interest) polygon pro výpočet statistických ukazatelů. Graf časové řady, statistické ukazatele i odkazy v pravé části okna programu FDI_tool umožňují prozkoumat změnu hodnot indexů v libovolném časovém období. Výsledky statistických výpočtů, změny oproti předešlému dostupnému datu i samotný ROI polygon lze uložit stisknutím tlačítka Export.



Obr. 7: Druhé okno programu FDI_tool (zdroj: Autorka BP, 2020)

8 DISKUZE

Výsledný program lze dále rozvíjet dle potřeb uživatelů. Jedním z uživatelsky přívětivějších řešení je použití Sandboxu pro lepší přenos virtuálního prostředí, které by zajistilo dostupnost verzí, v nichž je program vytvořen. Anaconda poskytuje možnost přenosu virtuálního prostředí pomocí serializačního formátu YAML. V něm jsou uvedeny potřebné moduly a knihovny pro spuštění a běh aplikace, které se stahují z úložiště Anaconda Cloud. Dostupnost verzí tedy závisí na tomto poskytovateli.

Aplikaci lze také více optimalizovat, případně využít paralelního zpracování pro zvýšení rychlosti procesů. Programový kód byl napsán bez předchozích zkušeností s použitými balíčky a s minimální znalostí jazyka Python a OOP. Zásahy do zdrojového kódu aplikace jsou možné, neboť je volně přístupný a editovatelný. V případě výskytu chyb knihovny GDAL – při maskování satelitních snímků vektorovou vrstvou ve formátu SHP (nerozpoznání podporovaného formátu TIFF) a při vytváření validního GeoJSON souboru (nenalezení souboru PROJ.db) je třeba program restartovat, případně ořezávat snímky vektorovou vrstvou postupně při každém dalším spuštění programu. Tyto chyby se vyskytují nepravidelně, a zatím stále nebyla odhalena jejich pravá příčina.

Mezi další možnosti rozšíření funkcionality patří přidání většího množství vegetačních indexů, zapojení dalších družic nebo využití neuronových sítí pro zpracování radarových dat. Jazyk Python je navíc velmi snadno propojitelný s jazykem R, čehož se v praxi často využívá. Zpracování dat by tak bylo rychlejší, efektivnější a nabízelo by více možností testování hypotéz a odhalování disturbance lesních porostů s vyšší mírou přesnosti. V případě zvýšení rychlosti je možné doplnit výpočty statistických ukazatelů pro jednotlivé segmenty vrstvy. Na jednom snímku by tak bylo možno typizovat lesní porosty např. na základě druhového složení a pro každý takový typizovaný segment zobrazit zmíněné statistiky.

Na snímcích z optických senzorů se běžně vyskytuje oblačnost znemožňující správné odečtení dat. Existují algoritmy a postupy na jejich potlačení nebo odstranění. Jedním z nich je poskládání snímků za co nejkratší časové období do bezoblačné mozaiky. Tuto metodu využívá například Ústav pro hospodářskou úpravu lesů nebo Evropská kosmická agentura (ESA) v projektu GlobCover. Značně jednodušší možností je identifikace a následné zvýraznění oblačnosti a výpočet procentuálního pokrytí snímku v dané oblasti, díky čemuž by si byl uživatel dobře vědom výše rizika spojeného s použitím dat do dalších analýz.

9 ZÁVĚR

Cílem této bakalářské práce bylo vytvořit funkční nástroj za účelem (polo)automatizace procesu identifikace disturbance lesních porostů na základě dat z družic Sentinel. Tento nástroj je k dispozici na GitHubu, jehož odkaz je mimo jiné na webových stránkách vytvořených za účelem prezentace této práce. Součástí nástroje je rovněž uživatelská příručka ve formátu PDF.

V teoretické části jsou podrobně rozepsány použité metody, data i programy k vytvoření nástroje. Následuje rozbor jejich příležitostí i omezení. Tato část je klíčová pro pochopení dalších postupů při tvorbě výsledné aplikace. Po skončení této etapy bylo navrženo grafické uživatelské prostředí a stanovila se stěžejní funkcionality programu. V průběhu samotné tvorby bylo řešeno mnoho vyvstalých problémů, ty hlavní z nich jsou rozebrány v diskuzi. Souběžně byl program doplňován o dílčí funkce, jež uživateli usnadní jeho použití. Mezi takové funkce patří interaktivní barevná stupnice, zaokrouhlení hodnot na tři desetinná místa, možnost přiblížení, oddálení nebo posunutí obrazu, vytvoření validního souboru GeoJSON či možnost nahrání vytvořené geometrie zpět do programu.

V závěru je třeba zmínit, že program byl testován na snímcích pokrývajících území Národního parku Krkonoše. Vegetační indexy byly zvoleny pro smíšené lesy mírného pásu a aplikace pro jiné podnebné oblasti může být zavádějící. V rámci střední Evropy a oblastí s podobným složením lesních porostů je však program použitelný nehledě na původní oblast testování.

POUŽITÁ LITERATURA A INFORMAČNÍ ZDROJE

Tištěné publikace

- ATZBERGER, C. Object-based retrieval of biophysical canopy variables using artificial neural nets and radiative transfer models. *Remote Sensing of Environment* [online]. 2004, str. 53-67 [cit. 2020-03-31].
- CLELAND, Elsa E., Isabelle CHUINE, Annette MENZEL, Harold A. MOONEY a Mark D. SCHWARTZ. Shifting plant phenology in response to global change. *TRENDS in Ecology and Evolution* [online]. 2007, 22(7) [cit. 2020-05-01]. DOI: <https://doi.org/10.1016/j.tree.2007.04.003>.
- HAVIRA, Miroslav, Radek BAČE, Vojtěch ČADA a Miroslav SVOBODA. *Množství a výšková struktura odrostlého zmlazení v horském smrkovém lese Hrubého Jeseníku*. Zprávy lesnického výzkumu [online]. Praha: Česká zemědělská univerzita v Praze, Fakulta lesnická a dřevařská, 2017, 62(1), 33-41 [cit. 2020-06-02].
- CHO, M. A. & A. K. SKIDMORE. A new technique for extracting the red edge position from hyperspectral data: The linear extrapolation method. *Remote Sensing of Environment* [online]. 2006, str. 181-193 [cit. 2020-03-31].
- KOLÁŘ, Jan, Lena HALOUNOVÁ a Karel PAVELKA. *Dálkový průzkum Země 10*. Praha: České vysoké učení technické, 1997. ISBN 80-01-01567-x.
- KRIŽOVÁ, Eva a Karol UJHÁZY. *Dynamika, stabilita a diverzita lesných ekosystémov*. Zvolen: TÚ Zvolen, 2007. ISBN 978-80-228-1821-6.
- LAUSCH, Angela, Stefan ERASMI, Douglas KING, Paul MAGDON a Marco HEURICH. *Understanding Forest Health with Remote Sensing – Part I—A Review of Spectral Traits, Processes and Remote-Sensing Characteristics*. Remote Sensing [online]. 2016, 8(12) [cit. 2020-06-02]. DOI: 10.3390/rs8121029. ISSN 2072-4292.
- LUKEŠ, Petr. *Hodnocení zdravotního stavu lesních porostů v České republice pomocí satelitních dat Sentinel-2* [online]. Brandýs nad Labem: Ústav pro hospodářskou úpravu lesů, 2018 [cit. 2020-06-02]. ISBN 978-80-88184-21-8.
- MAJASALMI, Titta. *The potential of Sentinel-2 data for estimating biophysical variables in a boreal forest: A simulation study* [online]. 2016 [cit. 2020-03-31].
- MERTON, Ray. *Monitoring community hysteresis using spectral shift analysis and the red-edge vegetation stress index* [online]. 1998 [cit. 2020-03-31].
- MÍCHAL, Igor. *Ekologická stabilita*. 2. rozš. vyd. Brno: Ministerstvo životního prostředí ČR, 1994. ISBN 80-853-6822-6.
- MILLAR, Constance I., Nathan L. STEPHENSON, Adam MORENO, Hubert HASENAUER, Rupert SEIDL. *Temperate forest health in an era of emerging megadisturbance*. Science [online]. 2015, 349(6250), 823-826 [cit. 2020-06-02]. DOI: 10.1126/science.aaa9933. ISSN 0036-8075.

NEUMANN, Mathias, Volker MUES, Adam MORENO, Hubert HASENAUER a Rupert SEIDL. Climate variability drives recent tree mortality in Europe. *Global Change Biology* [online]. 2017, 23(11), 4788-4797 [cit. 2020-06-02]. DOI: 10.1111/gcb.13724. ISSN 13541013.

POLÁK, Jiří, Antonín CARDA & Vojtěch MERUNKA. *Umění systémového návrhu: objektově orientovaná tvorba informačních systémů pomocí původní metody BORM*. Praha: Grada, 2003. Management v informační společnosti. ISBN 80-247-0424-2.

RAUTIAINEN, Miina, Janne HEISKANEN, Lars EKLUNDH, Matti MÖTTUS, Petr LUKEŠ a Pauline STENBERG. Ecological applications of physically based remote sensing methods. *Scandinavian Journal of Forest Research* [online]. 2010, 25(4), 325-339 [cit. 2020-05-01]. DOI: 10.1080/02827581.2010.497159.

STEINHAUSEN, Max J., Paul D. WAGNER, Balaji NARASIMHAN a Björn WASKE. Combining Sentinel-1 and Sentinel-2 data for improved land use and land cover mapping of monsoon regions. *Int. J. Appl. Earth Obs. Geoinformation* [online]. 2018, (73), 595-604 [cit. 2020-05-01].

STOKLASA, Milan a Petr FABIÁNEK. *Dálkový průzkum Země a pozemní šetření zdravotního stavu lesů v ČR*. Lesnická práce [online]. 1998, 77(10), 368-370 [cit. 2020-06-02]. ISSN 0322-9254.

SUMMERFIELD, Mark. *Python 3: výukový kurz*. Brno: Computer Press, 2010. ISBN 978-80-251-2737-7.

ZENÁHLÍKOVÁ, Jitka. *Přirozený vývoj horských lesů po rozsáhlých disturbancích*. Praha, 2012. Disertační práce. Česká zemědělská univerzita v Praze, Fakulta lesnická a dřevařská, Katedra pěstování lesů. Vedoucí práce Miroslav Svoboda.

ZHANG, Xiaoyang, Mark A. FRIEDL, Crystal B. SCHAAF, Alan H. STRAHLER, John C. F. HODGES, Feng GAO, Bradley C. REED a Alfredo HUETE. Monitoring vegetation phenology using MODIS. *Remote Sensing of Environment* [online]. 2003, 84(3), 471-475 [cit. 2020-05-01]. DOI: [https://doi.org/10.1016/S0034-4257\(02\)00135-9](https://doi.org/10.1016/S0034-4257(02)00135-9).

Elektronické zdroje

Anaconda Cloud [online]. [cit. 2020-03-31]. Dostupné z: <https://anaconda.org/>

Fiona 1.8.13.post1 [online]. [cit. 2020-03-31]. Dostupné z: <https://pypi.org/project/Fiona/>

ICP Forests [online]. [cit. 2020-06-02]. Dostupné z: <http://icp-forests.net/>

ITnetwork.cz: Návrh: Objektově orientované programování a evoluce vývoje softwaru [online]. [cit. 2020-03-31]. Dostupné z: <https://www.itnetwork.cz/navrh/objektove-orientovane-programovani-a-evoluece-vyvoje-softwaru>

ITnetwork.cz: Python: Lekce 2 - Proměnné, základní datové typy a funkce [online]. [cit. 2020-03-31]. Dostupné z: <https://www.itnetwork.cz/python/zaklady/python-tutorial-promenne-zakladni-datove-typy-a-funkce>

ISSUE 1686: diskuzní fórum GitHub v sekci mapbox / rasterio [online]. [2020-06-03].
Dostupné z: <https://github.com/mapbox/rasterio/issues/1686>

JetBrains [online]. [cit. 2020-03-31]. Dostupné z: <https://www.jetbrains.com/>

PyQt5 5.14.1 [online]. [cit. 2020-03-31]. Dostupné z: <https://pypi.org/project/PyQt5/>

PyQtGraph: Scientific Graphics and GUI Library for Python [online]. [cit. 2020-03-31].
Dostupné z: <http://www.pyqtgraph.org/>

Python [online]. [cit. 2020-03-31]. Dostupné z: <https://www.python.org>

Python: py.cz [online]. [cit. 2020-03-31]. Dostupné z: <https://www.py.cz/>

Rasterio [online]. [cit. 2020-03-31]. Dostupné z:
<https://rasterio.readthedocs.io/en/latest/>

Rasterstats [online]. [cit. 2020-03-31]. Dostupné z:
<https://pythonhosted.org/rasterstats/>

Sentinel 1 – Datové specifikace. *CollGS: Data z družic Sentinel v ČR* [online]. [cit. 2020-03-31]. Dostupné z: <https://collgs.czechspaceportal.cz/sentinel-1-datove-specifikace/>

Sentinel 2 – Datové specifikace. *CollGS: Data z družic Sentinel v ČR* [online]. [cit. 2020-03-31]. Dostupné z: <https://collgs.czechspaceportal.cz/sentinel-2-datove-specifikace/>

Stack Overflow [online]. [cit. 2020-03-31]. Dostupné z:
<https://insights.stackoverflow.com/survey/2019>

Terminologický slovník Výzkumného ústavu geodetického, topografického a kartografického, v. v. i. [online]. [cit. 2020-03-31]. Dostupné z:
<https://www.vugtk.cz/slovník/>

Ústav pro hospodářskou úpravu lesů: Katalog rychlých informací [online].
[cit. 2020-03-31]. Dostupné z: <http://www.uhul.cz/rychle-informace/>

PŘÍLOHY

SEZNAM PŘÍLOH

Volné přílohy

- Příloha 1 Poster k bakalářské práci
- Příloha 2 CD

Obsah CD

- Adresář s výsledným nástrojem (FDI)
- Dokumentace k programu
- Video návod
- Webové stránky